MX887030A/31A/32A/33A/40A/50A
Short Range Wireless Applications
Operation Manual

10th Edition

- **For safety and warning information, please read this manual before attempting to use the equipment.**
- **Additional safety and warning information is provided in the MT8870A Universal Wireless Test Set Operation Manual. Please refer to this document before using the equipment.**
- **Keep this manual with the equipment.**

ANRITSU CORPORATION

Document No.: M-W3617AE-10.0

Safety Symbols

To prevent the risk of personal injury or loss related to equipment malfunction, Anritsu Corporation uses the following safety symbols to indicate safety-related information. Ensure that you clearly understand the meanings of the symbols BEFORE using the equipment. Some or all of the following symbols may be used on all Anritsu equipment. In addition, there may be other labels attached to products that are not shown in the diagrams in this manual.

Symbols used in manual

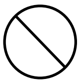
⚠ DANGER This indicates a very dangerous procedure that could result in serious injury or death if not performed properly.

⚠ WARNING This indicates a hazardous procedure that could result in serious injury or death if not performed properly.

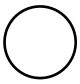
⚠ CAUTION This indicates a hazardous procedure or danger that could result in light-to-severe injury, or loss related to equipment malfunction, if proper precautions are not taken.

Safety Symbols Used on Equipment and in Manual

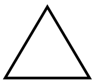
The following safety symbols are used inside or on the equipment near operation locations to provide information about safety items and operation precautions. Ensure that you clearly understand the meanings of the symbols and take the necessary precautions BEFORE using the equipment.



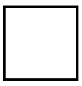
This indicates a prohibited operation. The prohibited operation is indicated symbolically in or near the barred circle.



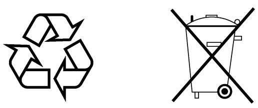
This indicates an obligatory safety precaution. The obligatory operation is indicated symbolically in or near the circle.



This indicates a warning or caution. The contents are indicated symbolically in or near the triangle.



This indicates a note. The contents are described in the box.



These indicate that the marked part should be recycled.

MX887030A/31A/32A/33A/40A/50A
Short Range Wireless Applications
Operation Manual

25 July 2012 (First Edition)
19 January 2018 (10th Edition)

Equipment Certificate

Anritsu Corporation guarantees that this equipment was inspected at shipment and meets the published specifications.

Anritsu Warranty

- During the warranty period, Anritsu Corporation will repair or exchange this software free-of-charge if it proves defective when used as described in the operation manual.
- The warranty period is 6 months from the purchase date.
- The warranty period after repair or exchange will remain 6 months from the original purchase date, or 30 days from the date of repair or exchange, depending on whichever is longer.
- This warranty does not cover damage to this software caused by Acts of God, natural disasters, and misuse or mishandling by the customer.

In addition, this warranty is valid only for the original equipment purchaser. It is not transferable if the equipment is resold.

Anritsu Corporation shall assume no liability for injury or financial loss of the customer due to the use of or a failure to be able to use this equipment.

Anritsu Corporation Contact

In the event of this equipment malfunctions, contact an Anritsu Service and Sales office. Contact information can be found on the last page of the printed version of this manual, and is available in a separate file on the PDF version.

Notes On Export Management

This product and its manuals may require an Export License/Approval by the government of the product's country of origin for re-export from your country.
Before re-exporting the product or manuals, please contact Anritsu to confirm whether they are export-controlled items or not.
When disposing of export-controlled items, the products/manuals must be decommissioned/shredded to prevent unlawful military use.

Software End-User License Agreement (EULA)

Please read this Software End-User License Agreement (hereafter this EULA) carefully before using (includes executing, copying, registering, etc.) this software (includes programs, databases, scenarios, etc., used to operate, set, etc., Anritsu electronic equipment). By reading this EULA and using this software, you are agreeing to be bound by the terms of its contents and Anritsu Corporation (hereafter Anritsu) hereby grants you the right to use this Software with the Anritsu-specified equipment (hereafter Equipment) for the purposes set out in this EULA.

1. Grant of License and Limitations

1. Regardless of whether this Software was purchased from or provided free-of-charge by Anritsu, you agree not to rent, lease, lend, or otherwise distribute this Software to third parties and further agree not to disassemble, recompile, reverse engineer, modify, or create derivative works of this Software.
2. You may make one copy of this Software for backup purposes only.
3. You are not permitted to reverse engineer this software.
4. This EULA allows you to install one copy of this Software on one piece of Equipment.

2. Disclaimers

To the extent not prohibited by law, in no event shall Anritsu be liable for personal injury, or any incidental, special, indirect or consequential damages whatsoever, including, without limitation, damages for loss of profits, loss of data, business interruption or any other commercial damages or losses, arising out of or related to your use or inability to use this Software.

3. Limitation of Liability

a. If a fault (bug) is discovered in this Software, preventing operation as described in the operation manual or specifications whether or not the customer uses this software as described in the manual, Anritsu shall at its own discretion, fix the bug, or exchange the software, or suggest a workaround, free-of-charge. However, notwithstanding the above, the following items shall be excluded from repair and warranty.
   - i) If this Software is deemed to be used for purposes not described in the operation manual or specifications.
   - ii) If this Software is used in conjunction with other non-Anritsu-approved software.
   - iii) Recovery of lost or damaged data.
   - iv) If this Software or the Equipment has been modified, repaired, or otherwise altered without Anritsu's prior approval.
   - v) For any other reasons out of Anritsu's direct control and responsibility, such as but not limited to, natural disasters, software virus infections, etc.

b. Expenses incurred for transport, hotel, daily allowance, etc., for on-site repairs by Anritsu engineers necessitated by the above faults shall be borne by you.

c. The warranty period for faults listed in article 3a above covered by this EULA shall be either 6 months from the date of purchase of this Software or 30 days after the date of repair, whichever is longer.

4. **Export Restrictions**

You may not use or otherwise export or re-export directly or indirectly this Software except as authorized by Japanese and United States law. In particular, this software may not be exported or re-exported (a) into any Japanese or US embargoed countries or (b) to anyone on the Japanese or US Treasury Department's list of Specially Designated Nationals or the US Department of Commerce Denied Persons List or Entity List. By using this Software, you warrant that you are not located in any such country or on any such list. You also agree that you will not use this Software for any purposes prohibited by Japanese and US law, including, without limitation, the development, design and manufacture or production of missiles or nuclear, chemical or biological weapons of mass destruction.

5. **Termination**

Anritsu shall deem this EULA terminated if you violate any conditions described herein. This EULA shall also be terminated if the conditions herein cannot be continued for any good reason, such as violation of copyrights, patents, or other laws and ordinances.

6. **Reparations**

If Anritsu suffers any loss, financial or otherwise, due to your violation of the terms of this EULA, Anritsu shall have the right to seek proportional damages from you.

7. **Responsibility after Termination**

Upon termination of this EULA in accordance with item 5, you shall cease all use of this Software immediately and shall as directed by Anritsu either destroy or return this Software and any backup copies, full or partial, to Anritsu.

8. **Dispute Resolution**

If matters of dispute or items not covered by this EULA arise, they shall be resolved by negotiations in good faith between you and Anritsu.

9. **Court of Jurisdiction**

This EULA shall be interpreted in accordance with Japanese law and any disputes that cannot be resolved by negotiation described in Article 8 shall be settled by the Japanese courts.

Cautions against computer virus infection

- Copying files and data
  Only files that have been provided directly from Anritsu or generated using Anritsu equipment should be copied to the instrument.
- Adding software
  Do not download or install software that has not been specifically recommended or licensed by Anritsu.
- Network connections
  Ensure that the network has sufficient anti-virus security protection in place.

CE Conformity Marking

Anritsu affixes the CE conformity marking to the following product(s) in accordance with Council Directive 768/2008/EC to indicate that they conform to the EMC and LVD directives of the European Union.

CE marking

1. Product Model

Software:

MX887030A WLAN 802.11b/g/a/n TX Measurement
MX887031A WLAN 802.11ac TX Measurement
MX887032A WLAN 802.11p TX Measurement
MX887033A WLAN 802.11ax TX Measurement
MX887040A Bluetooth TX Measurement
MX887050A Short Range Wireless Average Power and Frequency Measurement

2. Applied Directive and Standards

When the MX887030A, MX887031A, MX887032A, MX887033A, MX887040A or MX887050A Application software is installed in the MT8870A, the applied directive and standards of this software conform to those of the MT8870A main frame.

Note on Main Frame

Contact Anritsu for the latest information about the MT8870A Universal Wireless Test Set to be used with the MX887030A, MX887031A, MX887032A, MX887033A, MX887040A or MX887050A.

RCM Conformity Marking

Anritsu affixes the RCM mark to the following product(s) in accordance with the regulations to indicate that they conform to the EMC framework of Australia/New Zealand.

RCM marking

1. Product Model

Software:

MX887030A WLAN 802.11b/g/a/n TX Measurement
MX887031A WLAN 802.11ac TX Measurement
MX887032A WLAN 802.11p TX Measurement
MX887033A WLAN 802.11ax TX Measurement
MX887040A Bluetooth TX Measurement
MX887050A Short Range Wireless Average Power and Frequency Measurement

2. Applied Directive and Standards

When the MX887030A, MX887031A, MX887032A, MX887033A, MX887040A or MX887050A Application software is installed in the MT8870A, the applied directive and standards of this software conform to those of the MT8870A main frame.

Note on Main Frame

Contact Anritsu for the latest information about the MT8870A Universal Wireless Test Set to be used with the MX887030A, MX887031A, MX887032A, MX887033A, MX887040A or MX887050A.

About This Manual

This operation manual explains the setup and operation of the short range wireless applications for the MT8870A Universal Wireless Test Set.

Table of Contents

1
2
3
4
5
6
7
Appendix
Index

Chapter 1　General Information

This chapter provides an overview of the MT8870A and the MX8870xxA short range wireless applications.

1.1 About this Manual

This manual provides setup and operational information on the following short range wireless (SRW) transmitter testing applications for the MT8870A Universal Wireless Test Set.

MX887030A: WLAN 802.11b/g/a/n Tx Measurement

MX887031A: WLAN 802.11ac Tx Measurement

MX887032A: WLAN 802.11p Tx Measurement

MX887033A: WLAN 802.11ax Tx Measurement

MX887040A: *Bluetooth* Tx Measurement

MX887040A-001: DLE Tx Measurement

MX887040A-002: 2LE Tx Measurement

MX887040A-003: BLR Tx Measurement

MX887050A: Short range wireless average power and frequency measurement

Note:

Information on the cellular application options for the MT8870A can be found in the manuals supplied with the cellular options.

1.2 Instrument Overview

The MT8870A Universal Wireless Test Set is a one-box test instrument that can be configured to support the testing of both short range wireless and cellular communication standards.

The MT8870A is a modular instrument. It can be configured with up to four TRx modules, each equipped with a vector signal generator and vector signal analyzer.

A range of cellular and short range wireless applications provide support for transmitter testing on a number of communication standards.

Optional vector signal generator waveform files provide support for both cellular and SRW receiver testing.

The key strength of the MT8870A is its flexibility. The user can configure the instrument with between one and four modules and can perform cellular and/or SRW testing (in parallel or in sequence) on up to four devices under test (DUT). As testing requirements change, the instrument can be reconfigured: modules can be added and options for new wireless standards can be enabled.

This document provides details of the SRW transmitter testing applications. Refer to the MT8870A Universal Test Set Operation Manual for a more detailed description of the MT8870A

1.3 SRW Testing Overview

To make SRW measurements, the device under test (DUT) is placed into a state in which it transmits a stream of wireless packets (*Bluetooth* or WLAN) to the MT8870A. A Vector Signal Analyzer (VSA) within each MU887000A module captures and analyzes the received data and displays the requested measurement results in the user's application. Anritsu supplies the CombiTest and CombiView PC applications with the MT8870A but users may wish to use their own application instead.

The MT8870A also includes a Vector Signal Generator tool that can be used to perform receiver sensitivity testing when loaded with the appropriate SRW waveform files. Refer to the MU887000A TRX Test Module Operation Manual or the SRW Waveforms Manual for details of Rx testing.

1.4 MX8870xxA Features

- Supports WLAN Tx testing in compliance with IEEE specifications for 802.11b/g/a/n transmissions.
- Supports WLAN Tx testing in compliance with IEEE specifications for 802.11ac transmissions.
- Supports WLAN Tx testing in compliance with IEEE specifications for 802.11p transmissions.
- Supports WLAN Tx testing in compliance with IEEE specifications for 802.11ax transmissions.
- Supports two *Bluetooth* test modes: "SIG-Compliant" mode in which only those measurements that are compatible with the payload type are returned, and "Speed Test" mode in which results for all enabled measurements are returned regardless of the packet payload.
- Supports continuous wave power and frequency transmitter testing.

1.5 MX8870xxA Options

The SRW applications can be ordered with the instrument or added later.

If ordered later, a dedicated Utility Tool is used to enable the application by entering an activation code. Full details of this tool and the activation procedure are provided in Chapter 8 of the MU887000A TRX Test Module Operation Manual.

1.6 CombiView Overview

CombiView is a PC software application used to control the MT8870A Universal Wireless Test Set from a remote computer.

Short range wireless functionality is available within CombiView through the SRW applet. Once installed, and on the condition that the appropriate short range wireless option is detected, the SRW applet enables users to configure *Bluetooth* and WLAN measurements and display the results in both graphical and numerical forms.

Refer to the CombiView Operation Manual for full details of CombiView.

1.7 Conventions used in this Manual

Item	Convention
DUT	The term DUT is used to refer to the Device Under Test.
>	A chevron (>) may be used to indicate that you should select the items or keys in sequential order.
CombiView	The titles of windows and dialog boxes display in bold.
MX8870xxA	The term MX8870xxA is used throughout this manual when referring generically to all SRW applications for the MT8870A.
applet	Applets are installed in the CombiView framework to provide specific measurement or signal generator functions. Applets are available for Short Range Wireless, Cellular, and Vector Signal Generator.

Chapter 2 Principles of Operation

This chapter explains some key concepts that it is important to understand when using the MT8870A to make SRW measurements.

2.1 Data Capture

To make SRW transmitter measurements, the device under test (DUT) is placed into a state in which it transmits a stream of wireless packets (*Bluetooth* or WLAN) to the MT8870A.

The DUT streams packets to the MT8870A until a command is issued to stop from the chipset vendor's control software.

The vector signal analyzer (VSA) in the MU887000A module is configured to capture the data for a specific period of time or until a specified number of packets has been received. It analyzes the captured data and makes the measurement results available via its remote control interface.

2.2 Segmented Capture

One of the strengths of the MT8870A's SRW applications is the ability to perform segmented data capture and analysis.

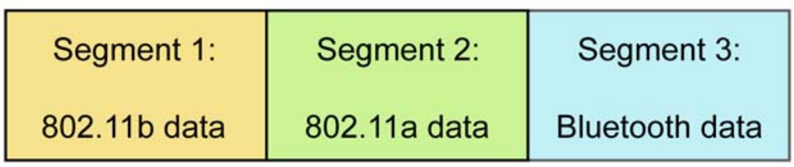
Segmented capture allows the user to make measurements on a continuous sequence of data segments transmitted from the DUT.

Each capture segment has a specified duration; the sum duration of all segments represents the total capture time.

Allowing for the partitioning of DUT transmissions into segments provides the flexibility for system configurations to change during a single capture.

If the DUT is able to produce a sequence of transmissions at various standards, the segmented capture feature of the VSA allows those parts of the resulting capture to be analyzed separately.

The user could, for example, configure a DUT supporting multiple wireless standards to transmit a stream of three segments to the MT8870A: a first segment of 802.11b data, a second segment of 802.11a data, and a third segment of *Bluetooth* data.



Segmented capture can also be used to make a stream of transmissions at the same communication standard but under different conditions. The user could, for example, configure the DUT to transmit three 802.11b segments to the DUT: a first segment at 1 Mbps, a second segment at 5.5 Mbps, and a third segment at 11 Mbps.

Segment 1: 802.11b data 1 Mbps	Segment 2: 802.11b data 5.5 Mbps	Segment 3: 802.11b data 11 Mbps

To capture and analyze the incoming data, the user must configure the system to match the incoming transmissions. Specifically, the user must set the wireless standard, the frequency of incoming transmissions, the expected power level, the number of packets in the segment, and the maximum segment time.

The system captures each segment and makes the required measurements. If the required measurements are known in advance, the measurement processing begins as soon as the data is captured. Alternatively, when capture is complete, the user can choose to display measurement results after selecting a specific packet for analysis.

Note:

In V1.00 the number of supported segment captures is limited to one.

2.3 Gates

The MT8870A's SRW applications use gates to obtain measurements from user-defined sections of a captured packet.

Gates have a "type" definition that can be set to "packet" or "user".

In "user" mode, the user defines the start position and stop positions. The start position is set by specifying a time offset from the start of a packet. The stop position is set by specifying the total gate width.

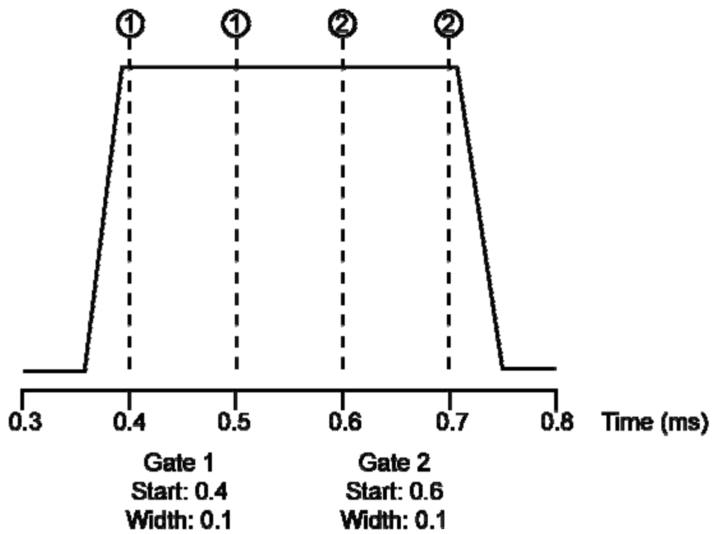
In "packet"' mode, the gate values are set automatically to encompass the entire packet.

Gate 1 cannot be disabled and is permanently set to "packet". It is assigned by default to all measurements that require a gate, but it can be un-assigned if required.

Eight gates are available in total, and measurements are obtained for the area within the specified gates.

Gates can be set to overlap if required and, by entering a minus value, can be set to begin before the start of the packet.

The figure below shows a packet with two gates.

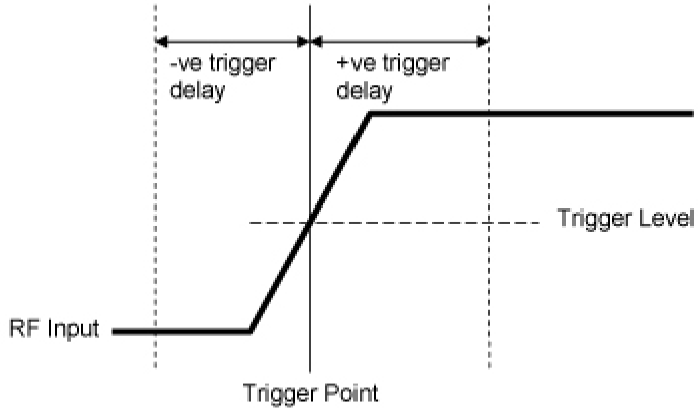


2.4 Triggering

2.4.1 Trigger Modes

There are two modes: “immediate” and “level”.

In Immediate mode the instrument starts to capture data immediately after receipt of an INIT:SRW command.



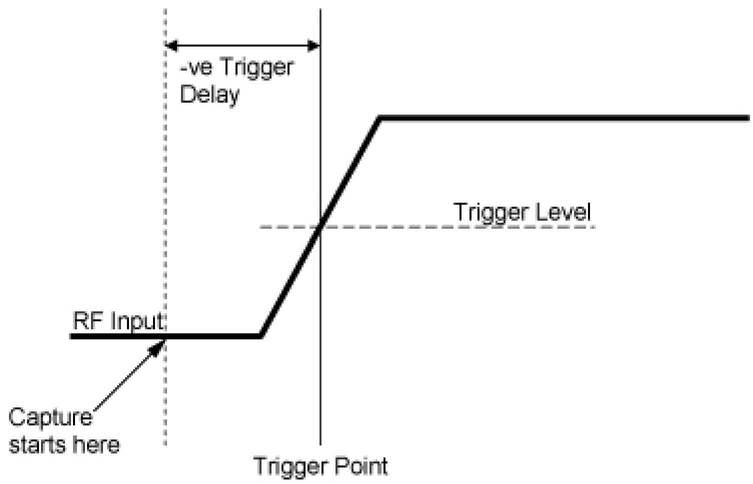
In Level mode the instrument starts to capture data when the RF input level exceeds the Trigger Level. The Trigger Level is specified relative to the expected input power level setting. By default it is –20 dB.

If, for example, the expected power level is –10 dBm and the trigger level is –20 dB, the capture starts when the input level exceeds –30 dBm.

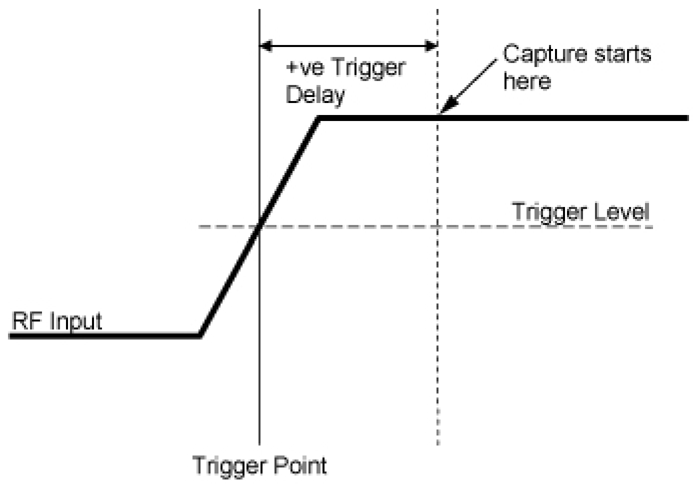
2.4.2 Trigger Delay

A Trigger Delay function can be used to adjust the point at which IQ data capture begins relative to the trigger point.

If the Trigger Delay has a negative value, the capture starts before the trigger point. This can be used to allow the whole of the power-on ramp to be captured and displayed.



If the Trigger Delay has a positive value, capture starts after the trigger point. This can be used to disregard the first of a sequence of packets.

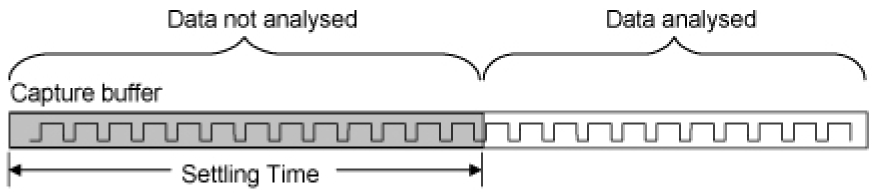


2.4.3 Settling Time

Settling time is an additional setting that has a similar effect to a positive trigger delay.

Settling time setting allows the first part of the captured data within a segment to be disregarded. This is useful for DUTs that require time to stabilize after a power level or frequency change.

By adjusting the settling time setting, the initial packets can be captured, but not included in the measurement.



Note:

It is important to understand the difference between a positive Trigger Delay setting and the Settling Time setting. With a positive trigger delay, IQ data is not captured during the time between the trigger and the expiry of the delay time. The settling time setting is applied to data that has been captured and stored in the IQ capture buffer. In effect it acts as a mask, preventing part of the captured data from being analyzed.

Typical command sequences are shown in the table below.

`:CONF:SRW:TRIG LEVEL`	Enable level triggering
`:CONF:SRW:TLEV -25`	Set the trigger level to –25 dB with respect to the expected input power level.
`:CONF:SRW:TDEL -5us`	Set up a trigger delay so that capture will begin 5 µs before the trigger point.
`:CONF:SRW:STIM 0`	Set the settling time to zero to ensure that data will be analyzed from the start of the capture.

2.5 *Bluetooth* Measurement Modes

The SRW *Bluetooth* application (MX887040A) provides two modes of measurement: “SIG-Compliant” mode and “Speed Test” mode.

2.5.1 SIG Compliant Mode

In SIG Compliant (or Standard) mode, the system returns only measurements that are compatible with the payload type of the captured packets. Measurements must be acquired for more than one packet type and this necessitates the capture of more than one segment or more than one capture if the DUT does not support sequence-based testing. The *Bluetooth* measurements that can be performed in SIG Compliant mode are shown in the tables below for each packet type. Supported measurements display a tick (check mark).

	Output Power	**Initial Carrier**	**Carrier Drift**	**Modulation Index**	**EDR Relative Tx Power**	**EDR Carrier and Modulation**
BR PRBS9	✓ DH1 DH3 DH5	✓ DH1, DH3 DH5				
BR 10101010	✓	✓	✓ DH1 DH3 DH5	✓ DH1 DH3 DH5		
BR 11110000	✓	✓		✓ DH1 DH3 DH5		
EDR PRBS9		✓			✓ 2-DH1, 3, 5 3-DH1, 3, 5	✓ 2-DH1, 3, 5 3-DH1, 3, 5

The following BLE SIG compliant transmitter measurements can be performed on each packet type.

	Output Power	**Modulation characteristics**	**Carrier freq offset and drift**
BLE PRBS9	✓		
BLE 10101010	✓	✓	✓
BLE 11110000	✓	✓	
2LE PRBS9	✓		
2LE 10101010	✓	✓	✓
2LE 11110000	✓	✓	
BLR S2 PRBS9	✓		
BLR S8 PRBS9	✓		
BLR S8 11111111	✓	✓	✓

2.5.2 Speed Mode

In Speed Test mode, the system returns results for all enabled measurements for a single capture segment regardless of the packet payload. The *Bluetooth* measurements that can be performed in Speed Test mode are shown in the table below for each packet type. Supported measurements display a tick (check mark).

	Output Power	Initial Carrier	Carrier Drift	Modulation Index	EDR Relative Tx Power	EDR Carrier & Modulation
BR PRBS9	✓	✓	✓	✓		
BR 10101010	✓	✓	✓	✓		
BR 11110000	✓	✓	✓	✓		
EDR PRBS9	✓	✓	✓	✓	✓	✓
Poll/Null	✓	✓	✓	✓		

Header exclusion mode is an extension to speed test mode that is available when the wireless standard selected for the segment is AUTOBT. In this mode the system does not rely on the packet header information being correct. Header exclusion mode is not available when the wireless standard is BT or BLE. If it is selected for these segments, or if the segment type is AUTOBT and a BLE packet is received, the system behaves as though speed test alone were selected.

Note:

Details of Speed Mode measurements that are not fully compliant with the *Bluetooth* specification are provided in chapter 4.

2.6 Measurement Results

SRW measurement results can be obtained for individual packets or for an average (a summary) of a number of packets specified for the capture segment.

The *Bluetooth* and WLAN standards set out different rules on the acceptability of averaged results.

The WLAN specification permits limit checking to be made using an average result for a number of packets.

The *Bluetooth* specification insists that each packet must pass individually (and summary results are therefore the logical AND of the status of all packets).

If the conditions of transmission (packet duration, trigger conditions, and the inter-packet spacing) are known, the capture length can be set to capture exactly the required number of packets. This may however be difficult and it is often easier to set the capture time to a value that is longer than the number of packets to be captured.

Chapter 3 WLAN Measurements

This chapter provides an overview of the WLAN transmitter measurements that can be made using the MX887030A, MX887031A, MX887032A and MX887033A options.

3.1 Transmit Power

Transmit Power is a measurement of the average power across a WLAN packet. Power is measured within a gate, typically positioned immediately after the rising edge and immediately before the falling edge.

The IEEE specification does not define the test conditions such as the packet length, data rate, or channel number. Instead, the maximum permissible output power is set by the regulatory body of the country or region.

The average power of an 802.11b device is typically +10 dBm to +16 dBm.

Some countries require control of output power to reduce transmitter power under certain conditions. Transmit power level control is a measurement of the average power across the WLAN packet taken after modifying (usually decreasing) the output power. The specification states that a radio capable of transmitting at greater than 100 mW must be able to switch back to 100 mW or less.

3.1.1 Peak Transmit Power

Peak power is the absolute maximum of the packet within the gate width period.

3.1.2 Average Transmit Power

The maximum allowable output power is specified by the regulatory body for the region in question. Refer to the table below.

		US	Europe	Japan
802.11b/g	2412 - 2472 MHz	30 dBm	20 dBm	10 dBm / MHz
	2484 MHz	-	-	10 dBm / MHz
802.11a	5150 - 5250 MHz	16 dBm	23 dBm	10 dBm / MHz
	5250 - 5350 MHz	23 dBm	23 dBm	10 dBm / MHz
	5470 - 5725 MHz	23 dBm	30 dBm	10 dBm / MHz
	5725 - 5825 MHz	29 dBm	-	-
	5825 MHz	30 dBm	-	-

As is the case for all power measurements, the average power is calculated from within the specified gate area only.

3.2 Crest Factor

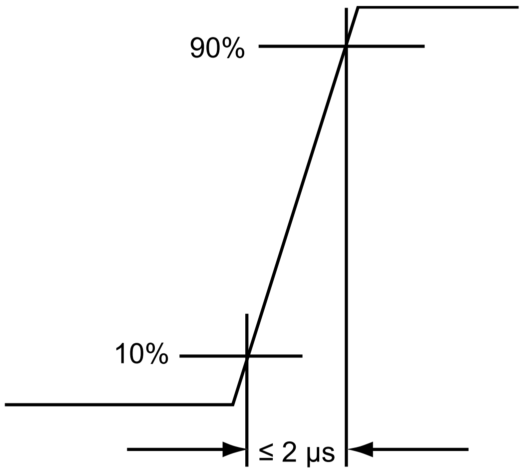
The crest factor represents the difference in dB between the peak and average power of the signal within the gate.

For 802.11b (DSSS), the crest factor is typically 3 to 4 dB. For OFDM transmissions (a, g, n, ac, p), the crest factor is typically 11 dB.

3.3 Power Ramp

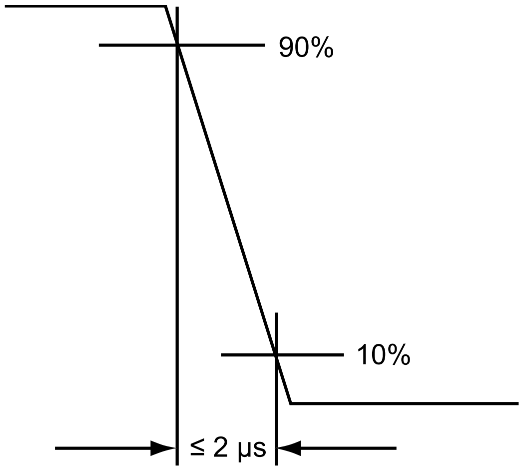
3.3.1 Transmit Power-On Ramp

For IEEE 802.11b, the specification states that the power-on ramp for the region between 10% and 90% of maximum power must not exceed 2 μs. This is shown in the figure below.



3.3.2 Transmit Power-Down Ramp

The IEEE 802.11b specification states that, as shown in the figure below, the power-down ramp for the region between 10% and 90% of maximum power must not exceed 2 μs. This is shown in the figure below.



3.4 Power Profile

Power Profile is a graphical measurement that returns either the average or peak power level as a function of time.

3.5 CCDF (OFDM only)

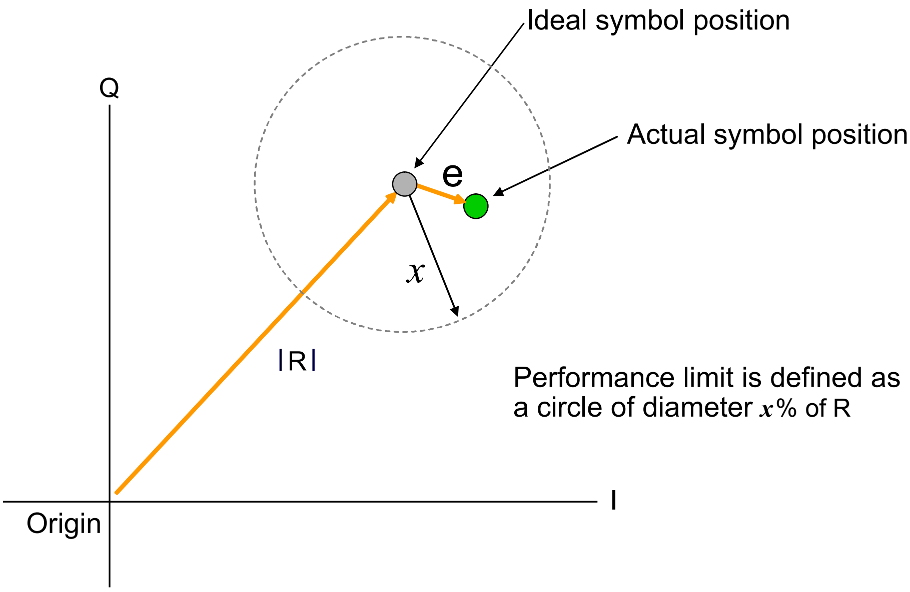
The Complementary Cumulative Distribution Function (CCDF) profile allows the user to ascertain the percentage of samples for which the power level specified by the markers is greater than the average power. The profile output contains 120 entries representing the power level (in dB) above the mean in 0.1 dB steps (0-12 dB above the mean). Each entry contains the percentage of time that the signal was less than this number of dB above the mean.

3.6 EVM

The IEEE 802.11 specification uses a metric called Error Vector Magnitude (EVM) as a measure of modulation quality. EVM is a measure of the difference between the ideal symbol position and the measured symbol position. The difference between these two positions is referred to as the error vector and is shown in the figure below by the distance “e”. EVM is an excellent measure of overall transmitter quality and a high EVM reading typically results in similarly high packet error rate.

The following points should be considered when performing IQ constellation / EVM measurements.

- The gate positions do not affect the EVM analysis.
- The burst captured must contain at least the specified number of chips/symbols.
- Independent analysis lengths for the DSSS and OFDM data rates are provided.



3.7 Spectral Flatness (OFDM Data Rates Only)

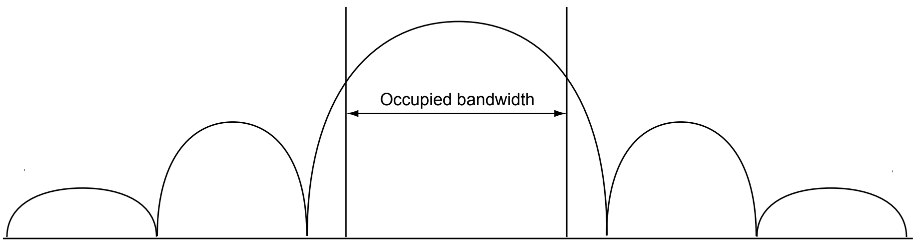
The spectral flatness measurement provides the user with a display of RF level vs sub-carrier over the measurement range. The IEEE 802.11a specification stipulates acceptable values for the average energy of the constellations in each of the sub-carriers. For lines -16 to -1 and 1 to 16 the average energy of any individual sub-carrier should deviate no more than ± 2 dB from their average energy of all 32 central sub-carriers. For lines -26 to -17 and 17 to 26 the average energy of the constellations in each of the spectral lines should deviate no more than +2/-4 dB from the average energy of spectral lines -16 to -1 and 1 to 16. The data for this test is measured over the 8 µs channel estimation period of the packet while all 52 sub-carriers are on.

Please use MCS0 signal for this measurement in 11ax Segment.

3.8 Occupied Bandwidth

Occupied bandwidth measures the frequency range within which 99% of the total power lies.

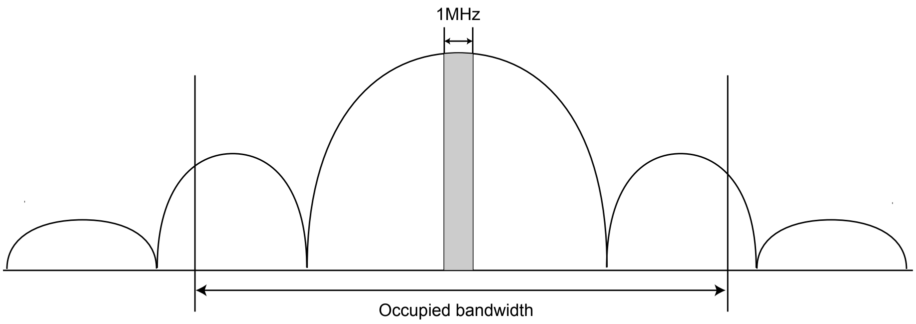
For products sold in some jurisdictions, the government requires all devices transmitting in the ISM band to meet a defined OBW value.



3.9 Power Spectral Density

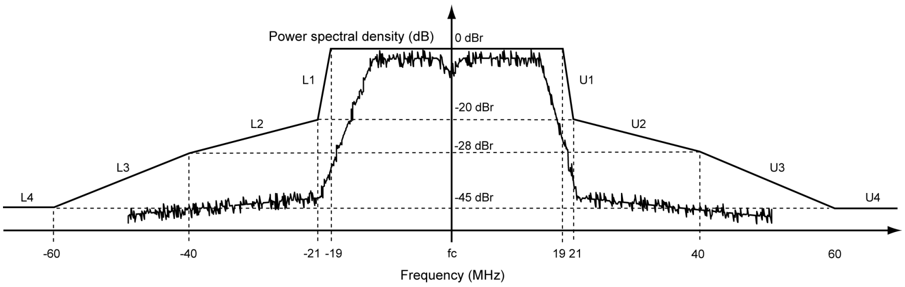
The power spectral density is the maximum power measured in any 1 MHz bandwidth within the occupied bandwidth of the signal.

For products sold in China, the Chinese government requires all devices transmitting in the ISM band to meet a defined PSD value.

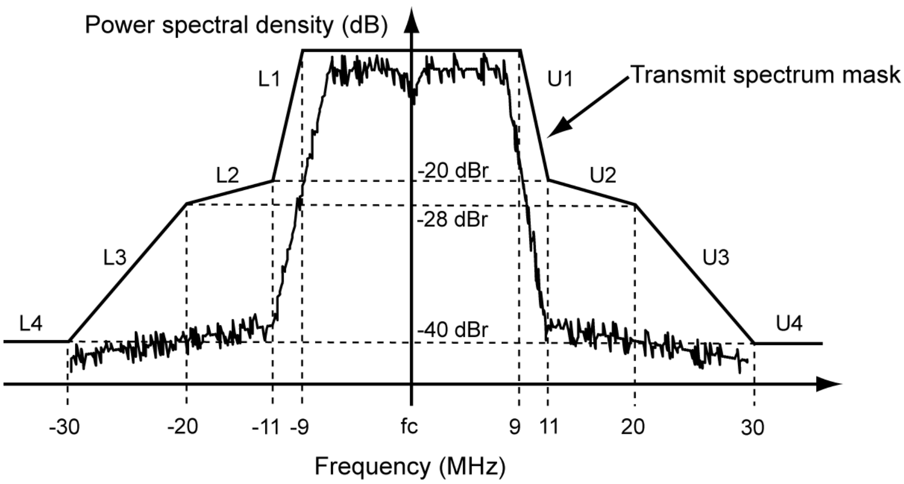


3.10 Spectrum Mask

The MT8870A applies the spectral mask for the selected WLAN standard. The results tables display the pass / fail status for each of the mask elements. The minimum gate time for spectrum updates is 50 µs. Longer gates have more spectrum samples giving the effect of point-to-point averaging.



Spectral Mask Compliance (40 MHz)



Spectral Mask Compliance (20 MHz)

3.11 Carrier Suppression

Carrier suppression is the level of the centre frequency tone relative to the modulation sideband with the highest power.

For 802.11b, this measurement requires an unscrambled 10101010 payload. If the DUT cannot be configured to transmit an unscrambled 10101010 data pattern, the MT8870A can estimate the carrier suppression based on the result of the IQ offset measurement.

The specification stipulates that carrier suppression at the centre frequency shall be at least 15 dB below the peak.

Any payload can be used when making carrier suppression measurements on an OFDM signal.

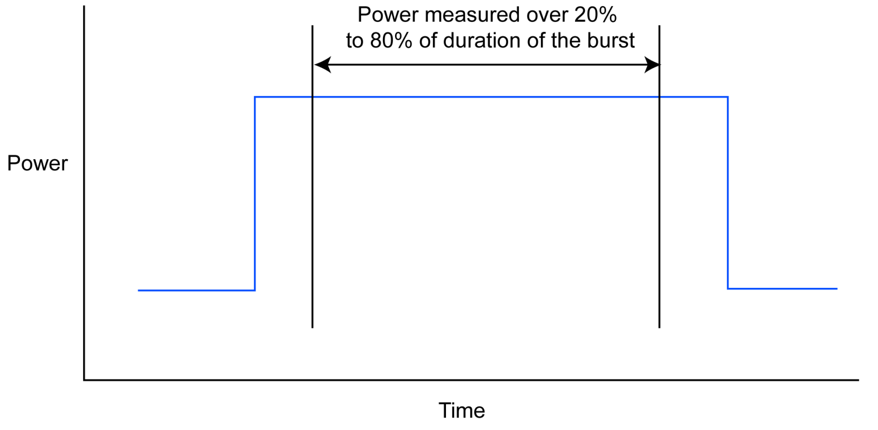
Chapter 4 Bluetooth Measurements

This chapter explains provides an overview of *Bluetooth* transmitter measurements that can be performed using the MX887040A option.

4.1 Basic Rate Tests

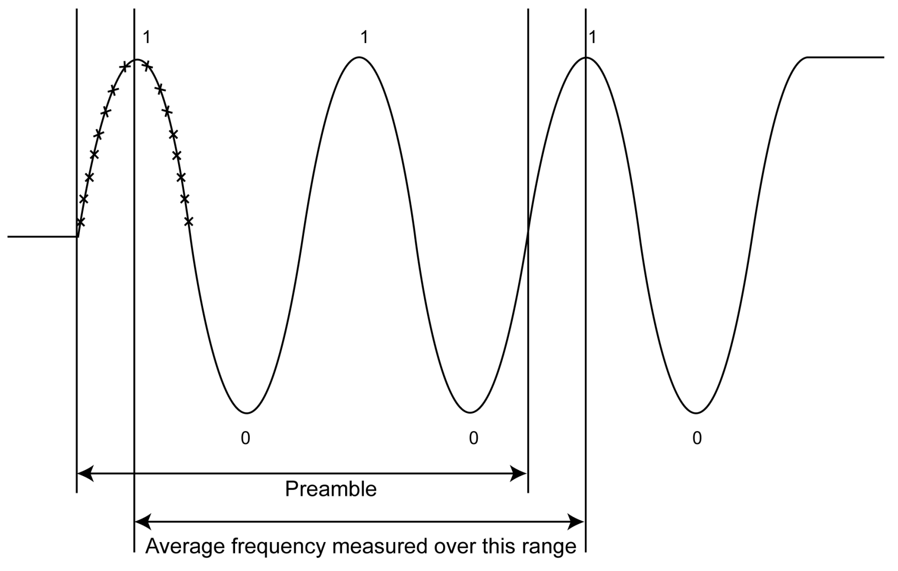
4.1.1 Output Power Test

The output power test measures the average power over 80% of the packet and peak RF-output power. Power is measured at low, middle, and high operating frequencies.



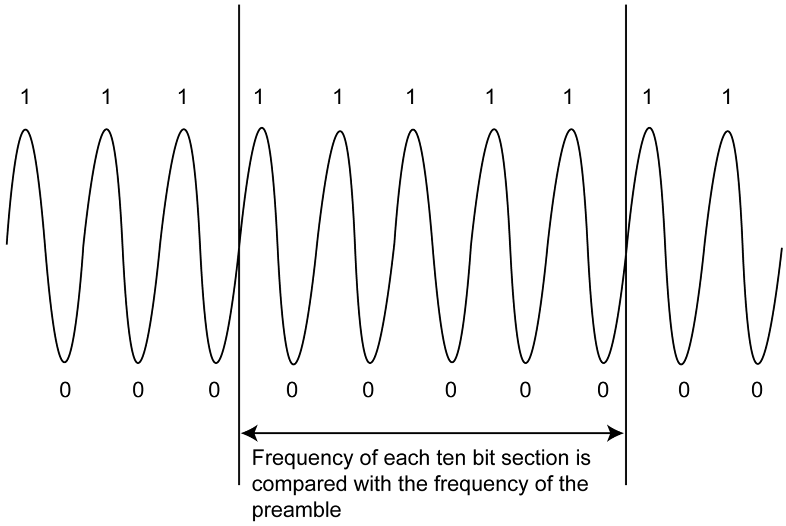
4.1.2 Initial Carrier Test

This test is used to verify the transmitter carrier frequency accuracy over the preamble at the start of the packet. A standard DH1 packet with pseudorandom bit sequence (PRBS) data is used.



4.1.3 Carrier Drift Test

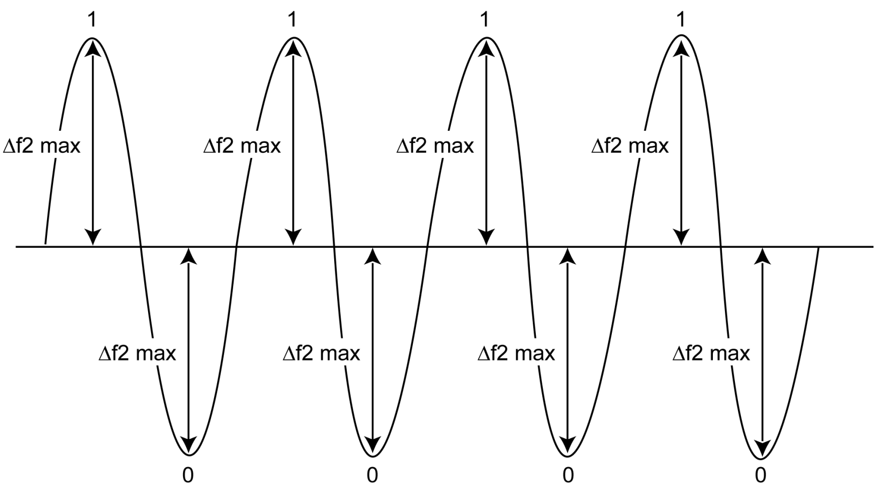
In the Carrier frequency drift test, the EUT is configured to generate DH1, DH3, or DH5 packets with a fixed 10101010 payload. A frequency drift measurement is performed over the length of the packet received.



4.1.4 Modulation Index Test

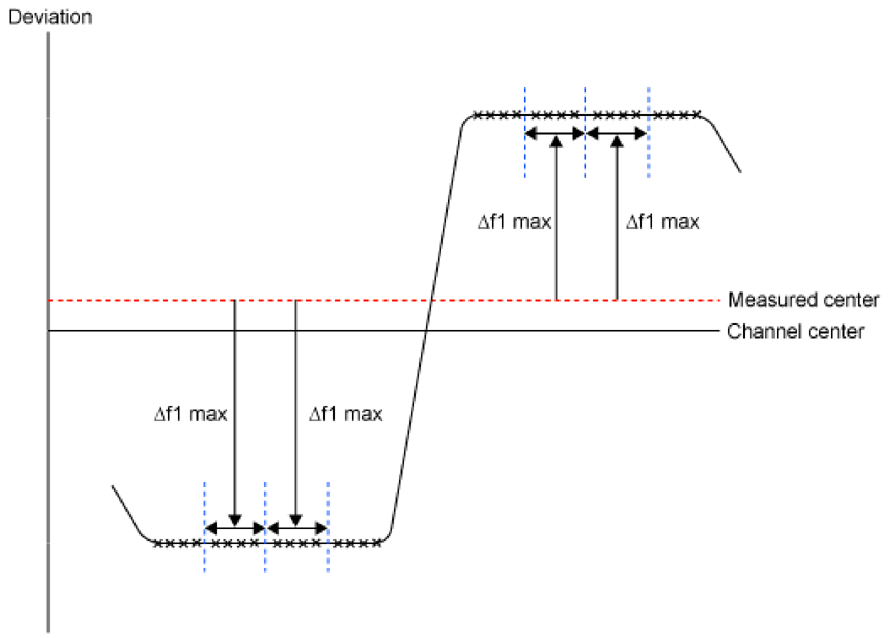
The modulation index is verified using 10101010 and 11110000 repeating 8-bit sequence in the payload. The combination of the two sequences checks the modulator performance and the pre-modulation filtering.

10101010 Modulation



Average of Δf2 max values calculated over 8 bursts.
99% of all Δf2 max values must be >140 kHz.

11110000 Modulation



A DHx packet with a 11110000 payload

The measurement of Δf2 max and Δf2 avg is derived by searching through the packet header for a 101 or 010 sequence, then measuring the centre symbol deviation and using this as the Δf2 max value.

A DHx packet with a PRBS9 payload

The measurement of Δf1 max and Δf1 avg is performed by searching through the whole packet, including header and payload, and looking for 111 or 000 bit sequences, then measuring the centre symbol deviation and using this as the Δf1 max value.

The measurement of Δf2 max and Δf2 avg is performed by searching through the whole packet including header and payload, for a 101 or 010 sequence, then measuring the centre symbol deviation and using this as the Δf2 max value.

A 2-DHx or 3-DHx packet with a PRBS9 payload

The measurement of Δf1 max and Δf1 avg is performed by searching through the packet header, and looking for 111 or 000 bit sequences, then measuring the centre symbol deviation and using this as the Δf1 max value.

The measurement of Δf2 max and Δf2 avg is performed by searching through the packet header, and looking for a 101 or 010 sequence, then measuring the centre symbol deviation and using this as the Δf2 max value. For any single packet, the following results shall be displayed;

- Δf1 max
- Δf1 avg
- Δf2 max
- Δf2 avg
- Δf2 avg / Δf1 avg
- % Δf2 max > 115 kHz

4.1.5 Adjacent Channel Power Test

The adjacent channel signal leakage of basic rate transmissions is measured up to ±5 channels from the channel on which the device under test is transmitting at maximum power. At ±2 channels the measured power must not exceed –20 dBm and for channels ±3, 4 or 5 it must not exceed –40 dBm

4.1.6 20 dB Bandwidth Test

Tests that the 20 dB bandwidth is less than 1.0 MHz for devices whose maximum power exceeds 0 dBm and 1.5 MHz for lower power devices. .

4.1.7 Frequency Range Test

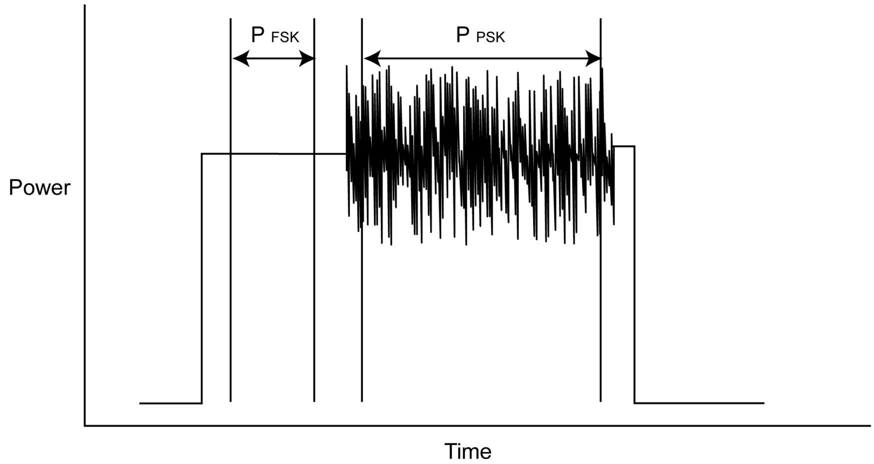
Ensures that transmissions outside the range 2.4 GHz to 2.4835 GHz do not exceed –30 dBm when measured in 100 kHz bandwidth.

4.2 Enhanced Data Rate Tests

The specification states that a radio capable of transmitting at greater than 100 mW must be able to switch back to 100 mW or less.

4.2.1 Relative Transmit Power Test

The EDR relative transmit power measurement ensures that the difference in average transmit power during the frequency modulated GFSK and phase modulated PSK parts of a packet is within limits.



4.2.2 Carrier and Modulation Test

This test verifies the transmitter carrier frequency stability and modulation accuracy. The test comprises of both a frequency measurement and a Differential Error Vector Magnitude (DEVM) measurement across the packet payload.

4.2.3 Differential Phase Test

In this measurement the DUT transmits a packet with a defined PRBS9 payload. The payload of the received packet is demodulated and compared with the defined ideal packet to give a resultant symbol error rate. The *Bluetooth* 2.1 specification stipulates that zero errors are detected in 99% of 100 packets transmitted. The *Bluetooth* test specification only requires this test to be performed on 2-DH1 and 3-DH1 packets on channel 0.

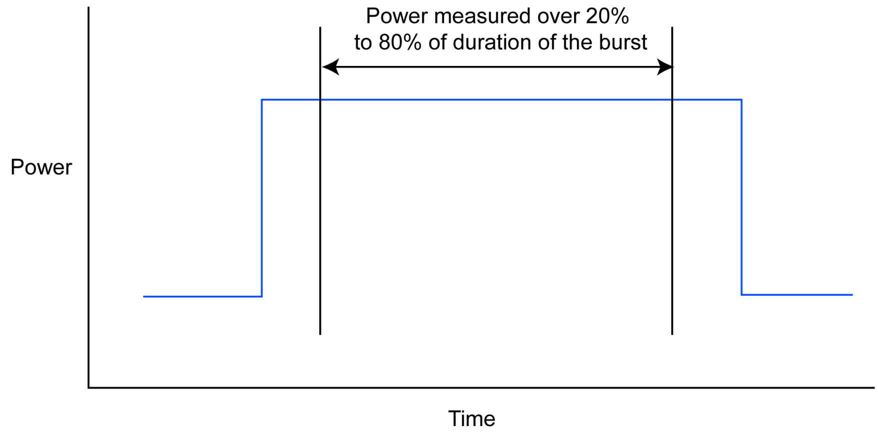
4.2.4 In-Band Spurious Emissions Test

Similar to the adjacent channel power measurement for basic rate transmissions, this test requires that the power measured in the immediately adjacent channels (±1) shall not exceed –26 dB with respect to the maximum power measured in the channel. At ±2 channels the measured power must not exceed –20 dBm and for channels ±3, 4 or 5 it must not exceed –40 dBm.

4.3 *Bluetooth* Low Energy Tests

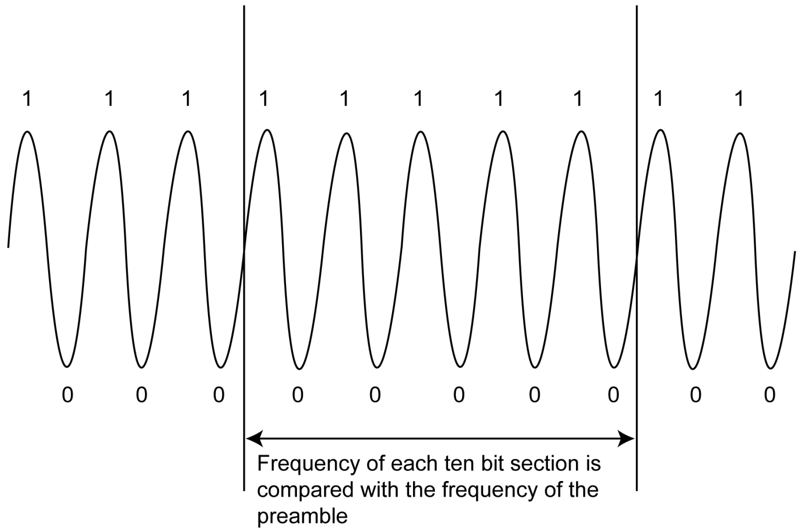
4.3.1 Output Power Test

The MT8870A transmits a test control message instructing the DUT to transmit reference test packets. The MT8870A measures the average power of the received packets over at least 20% to 80% of the duration of the burst.



4.3.2 Carrier Drift Test

The carrier drift test performs a frequency drift measurement over the length of the packet received. The carrier frequency offset is measured in the same manner as basic rate initial carrier test, but on the eight preamble bits in the low energy reference packet.

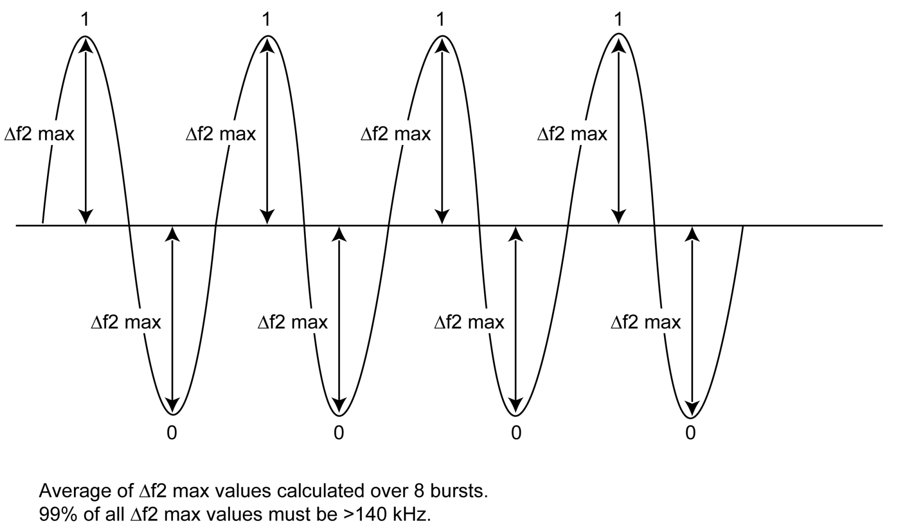


4.3.3 Modulation Index Test

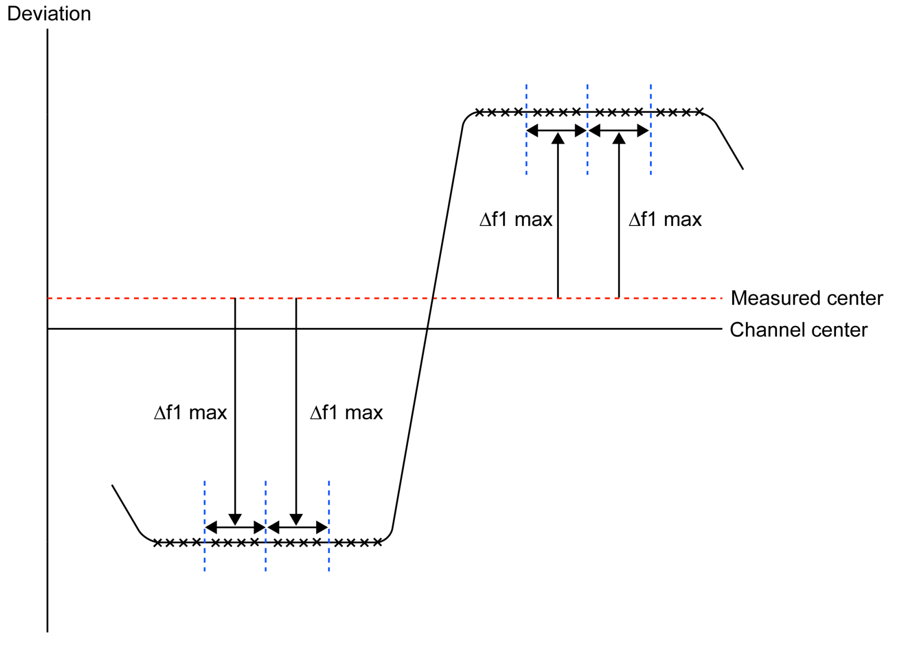
This test measures the modulation characteristics on the DUT output for each of the selected frequency ranges (LOW, MEDIUM and HIGH).

The MT8870A transmits a test control message over the RS232, USB, or 2-Wire interface instructing the DUT to transmit reference test packets with a four ones four zeros payload (11110000).

10101010 Modulation



11110000 Modulation



4.3.4 LE In-Band Spurious Emissions Test

At ±2 channels the measured power must not exceed –20 dBm and for channels ±3, 4 or 5 it must not exceed –30 dBm.

Chapter 5 Remote Operation

This chapter provides a reference to the remote commands that can be used to operate the MX8870xxA applications for the MT8870A.

5.1 Overview of Remote Operation

The MT8870A can be operated remotely over an Ethernet (or optional GPIB) interface.

This chapter provides details of the remote commands for the short range wireless applications. There are four optional applications and three additional options:

- MX887030A: WLAN 802.11b/g/a/n Tx Measurement
- MX887031A: WLAN 802.11ac Tx Measurement
- MX887032A: WLAN 802.11p Tx Measurement
- MX887033A: WLAN 802.11ax Tx Measurement
- MX887040A: Bluetooth Tx Measurement
- MX887040A-001: DLE Tx Measurement *MX887040A is required.
- MX887040A-002: 2LE Tx Measurement *MX887040A is required.
- MX887040A-003: BLR Tx Measurement *MX887040A is required.
- MX887050A: Short range wireless average power and frequency measurement

Refer to the MU88700A TRX Test Module Operation Manual for details of remote commands relating to the general operation and configuration of the MT8870A (those commands not specifically relating to any of the optional applications).

Before operating the instrument under remote control, users should be familiar with making measurements on the MT8870A and with the general operation of remote interfaces.

5.1.1 SCPI Overview

The remote commands used to control the MT8870A comply with the Standard Commands for Programmable Instruments (SCPI) protocol.

The SCPI Consortium sets out a defined syntax of commands and responses. The vertical and horizontal consistency of SCPI commands provides users with an intuitive and expandable programming environment in which compatibility is maintained across component applications and external equipment.

SCPI commands are constructed using a system of compound headers that allow complex commands to be built from a string of hierarchical elements. Each element in the string is separated from the next by a colon (:).

Most component elements within SCPI commands have short and long form styles. The short form sections are printed in upper case characters and the remaining sections are printed in lower case characters. A typical command is shown below.

:CONFigure:SRWireless:SEGMent:REMove

To execute this command, the user could send the entire string (as shown above) or just send the short form components:

:CONF:SRW:SEGM:REM

The command will be unrecognised if it is not sent in the exact short form or complete long form structure.

The query form of a command is created by appending a question mark to the end of the final element. A typical query command is shown below.

:CONFigure:SRWireless:CAPTure:TIME?

Square brackets ([]) are used as necessary in SCPI commands to enclose optional parameters.

5.1.2 Further Reading

The SCPI Consortium became part of the IVI Foundation in 2003. For a detailed introduction to the concept and principles of SCPI commands, visit http://www.ivifoundation.org/specifications/default.aspx and refer to the PDF document titled "SCPI-99".

5.2 List of Commands

5.2.1 Create and Manage Capture Segments

Function	Command / Query
Query only. Returns total available capture time and available free time.	:CONFigure:SRWireless:CAPTure:TIME?
Query only. Returns number of segments defined.	:CONFigure:SRWireless:SEGMent:NUMBer?
Set/query the active segment.	:CONFigure:SRWireless:SEGMent:ACTive/?
Append a new segment with a specified wireless standard.	:CONFigure:SRWireless:SEGMent:APPend
Remove a segment.	:CONFigure:SRWireless:SEGMent:REMove
Clear all segments	:CONFigure:SRWireless:SEGMent:CLEar
Specify how 20 MHz bandwidth OFDM legacy packets will be interpreted	:CONFigure:SRWireless:SEGMent:AUToid:OFDM:OVERride:BW20
Set/query the trigger mode for capturing the active segment	:CONFigure:SRWireless:TRIGger/?
Set/query the settling time for the active segment. (Time is from start of segment)	:CONFigure:SRWireless:STIMe/?
Query only. Returns wireless standard defined for active segment and user ID string.	:CONFigure:SRWireless:STANdard?
Set/query the RX frequency for the active segment.	:CONFigure:SRWireless:FREQuency/?
Set/query the expected power level for the active segment	:CONFigure:SRWireless:POWer/?
Set/query the duration of the active segment	:CONFigure:SRWireless:TIME/?
Set/query the number of packets to measure for the active segment	:CONFigure:SRWireless:PACKets/?
Set/query the number of packets to skip at the start of the active segment	:CONFigure:SRWireless:PACKets:SKIP/?
Set/query the trigger delay time for the active segment	:CONFigure:SRWireless:TDELay/?
Set/query the trigger level (relative to the expected power level) for the active segment	:CONFigure:SRWireless:TLEVel/?
Set/query the packet detector threshold level for the active segment	:CONFigure:SRWireless:PDTHreshold/?
Set/query the instrument receiver downconversion method for the active segment	:CONFigure:SRWireless:DCONversion/?

5.2.2 Gate Settings for the Active Segment

Function	Command / Query
Enable / Disable all gates / Read status of all gates	:CONFigure:SRWireless:GATE:ALL/?
Enable / Disable individual gates	:CONFigure:SRWireless:GATE:ENABle/?
Set/query the gate type	:CONFigure:SRWireless:GATE:TYPE/?
Set/query the gate start value relative to the start of the packet (when gate type = USER)	:CONFigure:SRWireless:GATE:STARt/?

Function	Command / Query
Set/query the gate width value (when gate type = USER)	:CONFigure:SRWireless:GATE:WIDTh/?

5.2.3 Power Profile for Active Segment

Function	Command / Query
Set/query the power profile mode	:CONFigure:SRWireless:PPRofile:MODE/?
Set/query the number of points in the power profile	:CONFigure:SRWireless:PPRofile:POINts/?
Set/query the start time of the power profile (relative to the packet rising edge).	:CONFigure:SRWireless:PPRofile:STARt/?
Set/query the stop time of the power profile (relative to the packet rising edge).	:CONFigure:SRWireless:PPRofile:STOP/?

5.2.4 Active Segment Frequency Deviation Profile

Function	Command / Query
Set/query the number of points per symbol	:CONFigure:SRWireless:DEViation:PPSYmbol/?

5.2.5 Capture Segment Interpretation

802.11b Packets

Function	Command / Query
Set/query the expected data rate of WLAN B packets in this segment	:CONFigure:SRWireless:WLB:DRATe/?
Set/query the expected filter type (gaussian ∣ root raised cosine)	:CONFigure:SRWireless:WLB:FTYPe/?
Set/query the root raised cosine filter alpha (applied if root raised cosine filter selected)	:CONFigure:SRWireless:WLB:FALPha/?
Set/query the gaussian filter BT (applied if gaussian filter selected)	:CONFigure:SRWireless:WLB:FBT/?

802.11a Packets

Function	Command / Query
Set/query the expected data rate of WLAN A packets in this segment	:CONFigure:SRWireless:WLA:DRATe/?

802.11g Packets

Function	Command / Query

Function	Command / Query
Set/query the expected data rate of WLAN G packets in this segment	:CONFigure:SRWireless:WLG:DRATe/?

802.11p Packets

Function	Command / Query
Set/query the expected data rate of WLAN P packets in this segment	:CONFigure:SRWireless:WLP:DRATe/?

802.11n Packets

Function	Command / Query
Set/query the expected PPDU format of the WLAN N packets in this segment	:CONFigure:SRWireless:WLN:PFORmat/?
Set/query the expected PPDU type of the WLAN N packets in this segment	:CONFigure:SRWireless:WLN:PTYPe/?
Set/query the expected MCS of the WLAN N packets in this segment	:CONFigure:SRWireless:WLN:MCS/?
Set/query the expected Guard Interval of the WLAN N packets in this segment	:CONFigure:SRWireless:WLN:GINTerval/?

802.11ac Packets

Function	Command / Query
Set/query the expected PPDU format of the WLAN AC packets in this segment	:CONFigure:SRWireless:WLAC:PTYPe/?
Set/query the expected MCS of the WLAN AC packets in this segment	:CONFigure:SRWireless:WLAC:MCS/?
Set/query the expected Guard Interval of the WLAN AC packets in this segment	:CONFigure:SRWireless:WLAC:GINTerval/?
Set/query whether the IEEE 802.11ac 80+80 frequency segments are separate or combined.	:CONFigure:SRWireless:WLAC:SPECtrum:C80_80/?

802.11ax Packets

Function	Command / Query
Set/query the expected PPDU format of the WLAN AX packets in this segment	:CONFigure:SRWireless:WLAX:PFORmat/?
Set/query the expected PPDU type of the WLAN AX packets in this segment	:CONFigure:SRWireless:WLAX:PTYPe/?
Set/query the expected MCS of the WLAN AX packets in this segment	:CONFigure:SRWireless:WLAX:MCS/?
Set/query the expected Guard Interval of the WLAN AX packets in this segment	:CONFigure:SRWireless:WLAX:GINTerval/?
Set/query the expected HE-LTF size of the WLAN AX packets in this segment	:CONFigure:SRWireless:WLAX:LSIZe/?

Bluetooth General

Function	Command / Query
Set/query the *Bluetooth* address for basic rate, EDR, and low energy	:CONFigure:SRWireless:BT:ADDRess/?

Bluetooth Basic Rate or EDR Packets

Function	Command / Query
Set/query the measurement mode: (“SIG standard” or “Speed Test”)	:CONFigure:SRWireless:BT:MODE/?
Set/query the expected packet type of the BT (BR / EDR) packets in this segment	:CONFigure:SRWireless:BT:PACKet/?
Set/query the expected payload type of the BT (BR / EDR) packets in this segment.	:CONFigure:SRWireless:BT:PAYLoad/?
Set/query the expected payload length of the BT (BR / EDR) packets in this segment	:CONFigure:SRWireless:BT:PLENgth/?

Bluetooth Low Energy Packets

Function	Command / Query
Set/query the measurement mode: (“SIG standard” or “Speed Test”)	:CONFigure:SRWireless:BLE:MODE/?
Set/query the expected payload type of the BT (LE) packets in this segment.	:CONFigure:SRWireless:BLE:PAYLoad/?
Set/query the expected BLE payload length.	:CONFigure:SRWireless:BLE:PLENgth
Set/query the expected BLE packet type.	:CONFigure:SRWireless:BLE:PACKet/?
Set/query the expected BLR (BLE coded) coding scheme.	:CONFigure:SRWireless:BLE:LRCoding/?

5.2.6 Active Segment Measurement Settings

General measurement settings (OFDM and DSSS packets)

Function	Command / Query
Set/query the percentage of transmitter power for which the bandwidth is calculated	:CONFigure:SRWireless:OBW:POWer/?
Add a frequency offset to the list defined for use when fetching spot spectrum measurements.	:CONFigure:SRWireless:WLAN:SPOT:FREQuency:ADD
Clear the list of frequency offsets defined for use when fetching spot spectrum measurements.	:CONFigure:SRWireless:WLAN:SPOT:FREQuency:CLEar
Set the window type used for spectrum analysis measurements in the active segment.	:CONFigure:SRWireless:SPECtrum:WINDow
Enable or disable full span spectral mask measurements.	:CONFigure:SRWireless:SELect:WLAN:MASK:FULL
Enable full span 802.11ac (80 MHz, 160MHz) measurements.	:CONFigure:SRWireless:SELect:WLAN:SPECtru

Function	Command / Query
	m:FULL
Set the standard used for 802.11b spectral mask measurements.	:CONFigure:SRWireless:WLB:SMASk
Set a target 'percentage of samples' value (on the CCDF graph y-axis) for which the corresponding power distribution measurement will be returned by:FETC:SRW:SUMM:WLAN:CCDF:PDIS?	:CONFigure:SRWireless:WLAN:CCDF:PSAMples

True MIMO

Associate an MU88700xA module's IP address with a TMIMOS segment	:CONFigure:SRWireless:SEGMent:TMIMo:IP
Connect to all true MIMO "S" modules using the IP addresses assigned in the TMIMOS segments and transfer configuration information.	:CONFigure:SRWireless:TMIMo:CONNect
Cancel the connections to all true MIMO "S" modules.	:CONFigure:SRWireless:TMIMo:DISConnect
Query the status of all connections to MU88700xA modules used for a true MIMO measurement.	:STATus:SRWireless:TMIMo:CONNect?

Composite MIMO

Function	Command / Query
Set the Composite MIMO reference data for the active segment	:CONFigure:SRWireless:WLAN:CMIMo:REFerence

EVM measurement settings for DSSS packets

Function	Command / Query
Set/query the DSSS EVM calculation method	:CONFigure:SRWireless:DSSS:CMEThod/?
Set/query the DSSS EVM analysis length mode	:CONFigure:SRWireless:DSSS:ALMode/?
Set/query the DSSS EVM analysis length in chips.	:CONFigure:SRWireless:DSSS:ALENgth/?
Set/query the DSSS chip clock frequency error analysis length	:CONFigure:SRWireless:DSSS:CCTLength/?

EVM measurement settings for OFDM packets

Function	Command / Query
Set/query the OFDM EVM analysis length mode	:CONFigure:SRWireless:OFDM:ALMode/?
Set/query the OFDM EVM analysis length	:CONFigure:SRWireless:OFDM:ALENgth/?
Set/query the OFDM frequency correction method	:CONFigure:SRWireless:OFDM:FCORrection/?
Set/query the OFDM clock error analysis length mode	:CONFigure:SRWireless:OFDM:CEAMode/?

Function	Command / Query
Set/query the OFDM clock error analysis length	:CONFigure:SRWireless:OFDM:CEALength/?
Set/query the OFDM channel estimation setting	:CONFigure:SRWireless:OFDM:CESTimation/?
Set/query the OFDM pilot tracking setting	:CONFigure:SRWireless:OFDM:PTRacking/?
Set/query the OFDM MIMO EVM correction for cross-stream leakage.	:CONFigure:SRWireless:OFDM:MIMO:EVM:CORRection/?

5.2.7 Active Segment Limit Checking

CW

Function	Command / Query
Set/query CW power upper limit	:CONFigure:SRWireless:CW:POWer:ULIMit/?
Set/query CW power lower limit	:CONFigure:SRWireless:CW:POWer:LLIMit/?
Set/query CW frequency offset limit	:CONFigure:SRWireless:CW:FREQuency:LIMit/?

WLAN

Function	Command / Query
Set/query WLAN Transmit power upper limit	:CONFigure:SRWireless:POWer:ULIMit/?
Set/query WLAN Transmit power lower limit	:CONFigure:SRWireless:POWer:LLIMit/?
Set/query WLAN Transmit power upper limit for each IEEE 802.11p power class.	:CONFigure:SRWireless:WLP:POWer:ULIMit/?
Set/query WLAN Transmit power lower limit for each IEEE 802.11p power class.	:CONFigure:SRWireless:WLP:POWer:LLIMit/?
Set/query WLAN Transmit power class (IEEE 802.11p)	:CONFigure:SRWireless:WLP:POWer:PCLass
Set/query WLAN power on ramp time upper limit	:CONFigure:SRWireless:PORamp:ULIMit/?
Set/query WLAN power down ramp time upper limit	:CONFigure:SRWireless:PDRamp:ULIMit/?
Set/query WLAN A EVM upper limits	:CONFigure:SRWireless:WLA:EVM:ULIMit/?
Set/query WLAN B EVM upper limits	:CONFigure:SRWireless:WLB:EVM:ULIMit/?
Set/query WLAN G EVM upper limits	:CONFigure:SRWireless:WLG:EVM:ULIMit/?
Set/query WLAN P EVM upper limits	:CONFigure:SRWireless:WLP:EVM:ULIMit/?
Set/query WLAN N EVM upper limits	:CONFigure:SRWireless:WLN:EVM:ULIMit/?
Set/query WLAN AC EVM upper limits	:CONFigure:SRWireless:WLAC:EVM:ULIMit/?
Set/query WLAN AX EVM upper limits	:CONFigure:SRWireless:WLAX:EVM:ULIMit/?
Set/query WLAN A (OFDM) symbol clock frequency tolerance limit (±)	:CONFigure:SRWireless:WLA:SCFTolerance:LIMit/?

Function	Command / Query
Set/query WLAN G (OFDM) symbol clock frequency tolerance limit (±)	:CONFigure:SRWireless:WLG:SCFTolerance:LIMit/?
Set/query WLAN P (OFDM) symbol clock frequency tolerance limit (±).	:CONFigure:SRWireless:WLP:SCFTolerance:LIMit/?
Set/query WLAN N (OFDM) symbol clock frequency tolerance limit (±)	:CONFigure:SRWireless:WLN:SCFTolerance:LIMit/?
Set/query WLAN AC (OFDM) symbol clock frequency tolerance limit (±)	:CONFigure:SRWireless:WLAC:SCFTolerance:LIMit/?
Set/query WLAN AX (OFDM) symbol clock frequency tolerance limit (±)	:CONFigure:SRWireless:WLAX:SCFTolerance:LIMit/?
Set/query WLAN A (OFDM) transmitter center frequency tolerance limit (±)	:CONFigure:SRWireless:WLA:TCFTolerance:LIMit/?
Set/query WLAN G (OFDM) transmitter center frequency tolerance limit (±)	:CONFigure:SRWireless:WLG:TCFTolerance:LIMit/?
Set/query WLAN P (OFDM) transmitter center frequency tolerance limit (±)	:CONFigure:SRWireless:WLP:TCFTolerance:LIMit/?
Set/query WLAN N (OFDM) transmitter center frequency tolerance limit (±)	:CONFigure:SRWireless:WLN:TCFTolerance:LIMit/?
Set/query WLAN AC (OFDM) transmitter center frequency tolerance limit (±)	:CONFigure:SRWireless:WLAC:TCFTolerance:LIMit/?
Set/query WLAN AX (OFDM) transmitter center frequency tolerance limit (±)	:CONFigure:SRWireless:WLAX:TCFTolerance:LIMit/?
Set/query WLAN B (DSSS) transmitter center frequency tolerance limit (±)	:CONFigure:SRWireless:WLB:TCFTolerance:LIMit/?
Set/query WLAN A (OFDM) transmitter center frequency leakage upper limit	:CONFigure:SRWireless:WLA:TCFLeakage:ULIMit/?
Set/query WLAN G (OFDM) transmitter center frequency leakage upper limit	:CONFigure:SRWireless:WLG:TCFLeakage:ULIMit/?
Set/query WLAN P (OFDM) transmitter center frequency leakage upper limit	:CONFigure:SRWireless:WLP:TCFLeakage:ULIMit/?
Set/query WLAN N (OFDM) transmitter center frequency leakage upper limit	:CONFigure:SRWireless:WLN:TCFLeakage:ULIMit/?
Set/query WLAN AC (OFDM) transmitter center frequency leakage upper limit	:CONFigure:SRWireless:WLAC:TCFLeakage:ULIMit/?
Set/query WLAN AX (OFDM) transmitter center frequency leakage upper limit	:CONFigure:SRWireless:WLAX:TCFLeakage:ULIMit/?
Set/query WLAN B (DSSS) RF carrier suppression lower limit	:CONFigure:SRWireless:WLB:RFCSuppress:LLIMit/?
Set/query WLAN B (DSSS) chip clock frequency tolerance limit (±)	:CONFigure:SRWireless:WLB:CCFTolerance:ULIMit/?
Set/query WLAN A (OFDM) spectral flatness limits	:CONFigure:SRWireless:WLA:SFLatness:LIMit/?
Set/query WLAN G (OFDM) spectral flatness limits	:CONFigure:SRWireless:WLG:SFLatness:LIMit/?
Set/query WLAN P (OFDM) spectral flatness limits	:CONFigure:SRWireless:WLP:SFLatness:LIMit/?
Set/query WLAN N (OFDM) spectral flatness limits	:CONFigure:SRWireless:WLN:SFLatness:LIMit/?
Set/query WLAN AC (OFDM) spectral flatness limits	:CONFigure:SRWireless:WLAC:SFLatness:LIMit/?
Set/query WLAN AX (OFDM) spectral flatness limits	:CONFigure:SRWireless:WLAX:SFLatness:LIMit/?

Function	Command / Query
Set/query WLAN power spectral density upper limit. (Applies to all wireless LAN standards)	:CONFigure:SRWireless:WLAN:PSD:ULIMit/?
Set/query WLAN occupied bandwidth upper limit (for 99.9 % of transmitted power).	:CONFigure:SRWireless:WLAN:OBW:ULIMit/?
WLAN B (DSSS) RF carrier suppression (from IQ Offset) lower limit.	:CONFigure:SRWireless:WLB:IQCSuppress:LLIMit
Define the spectral mask limits for WLAN IEEE 802.11a signals.	:CONFigure:SRWireless:WLA:SMASk:ULIMit
Define the spectral mask limits for WLAN IEEE 802.11ac signals.	:CONFigure:SRWireless:WLAC:SMASk:ULIMit
Define the spectral mask limits for WLAN IEEE 802.11ax signals.	:CONFigure:SRWireless:WLAX:SMASk:ULIMit
Define the spectral mask limits for WLAN IEEE 802.11b signals.	:CONFigure:SRWireless:WLB:SMASk:ULIMit
Define the spectral mask limits for WLAN IEEE 802.11g signals.	:CONFigure:SRWireless:WLG:SMASk:ULIMit
Define the spectral mask limits for WLAN IEEE 802.11p signals.	:CONFigure:SRWireless:WLP:SMASk:ULIMit
Define the spectral mask for an IEEE 802.11p power class.	:CONFigure:SRWireless:WLP:SMASk:PCLass:ULIMit
Set/query the IEEE 802.11p power class for spectral mask measurements.	:CONFigure:SRWireless:WLP:SMASk:PCLass
Define the spectral mask limits for WLAN IEEE 802.11n signals.	:CONFigure:SRWireless:WLN:SMASk:ULIMit

BLUETOOTH

Function	Command / Query
Set/query BT average output power lower limit (TRM/CA/01/C)	:CONFigure:SRWireless:BT:APOWer:LLIMit/?
Set/query BT average output power upper limit (TRM/CA/01/C)	:CONFigure:SRWireless:BT:APOWer:ULIMit/?
Set/query BT peak power upper limit (TRM/CA/01/C)	:CONFigure:SRWireless:BT:PPOWer:ULIMit/?
Set/query BT frequency range measurement limit (TRM/CA/04/C)	:CONFigure:SRWireless:BT:FRANge:LIMit/?
Set the BT adjacent power upper limits (TRM/CA/06/C)	:CONFigure:SRWireless:BT:ACPower:ULIMit
Set/query BT initial carrier frequency tolerance limit (TRM/CA/08/C)	:CONFigure:SRWireless:BT:ICFTolerance:LIMit/?
Set/query BT carrier frequency drift rate limit (TRM/CA/09/C)	:CONFigure:SRWireless:BT:DRATe:LIMit/?
Set/query BT carrier frequency drift limit (TRM/CA/09/C)	:CONFigure:SRWireless:BT:CDRift:LIMit/?
Set/query BT Δ f1 average lower limit (TRM/CA/07/C)	:CONFigure:SRWireless:BT:F1AVerage:LLIMit/?
Set/query BT Δ f1 average upper limit (TRM/CA/07/C)	:CONFigure:SRWireless:BT:F1AVerage:ULIMit/?
Set/query BT Δ f2 maximum lower limit (TRM/CA/07/C)	:CONFigure:SRWireless:BT:F2Maximum:LLIMit/?
Set/query BT Δ f2 average / delta f1 average lower limit (TRM/CA/07/C)	:CONFigure:SRWireless:BT:F21Ratio:LLIMit/?
Set/query BT EDR relative transmit power lower limit (TRM/CA/10/C)	:CONFigure:SRWireless:BT:ERTPower:LLIMit/?
Set/query BT EDR relative transmit power upper limit (TRM/CA/10/C)	:CONFigure:SRWireless:BT:ERTPower:ULIMit/?

Function	Command / Query
Set/query BT EDR carrier frequency stability limit (TRM/CA/11/C)	:CONFigure:SRWireless:BT:CFSTability:LIMit/?
Set/query BT EDR RMS DEVM upper limit (TRM/CA/11/C)	:CONFigure:SRWireless:BT:RDEVm:ULIMit/?
Set/query BT EDR Peak DEVM upper limit (TRM/CA/11/C)	:CONFigure:SRWireless:BT:PDEVm:ULIMit/?
Set/query BT EDR DEVM limit to be achieved by 99% of symbols (TRM/CA/11/C)	:CONFigure:SRWireless:BT:PCDevm:ULIMit/?
Set/query BT EDR guard interval (included with TRM/CA/11/C settings)	:CONFigure:SRWireless:BT:GUARd:LIMit/?
Set/query the BT EDR differential phase encoding packet error rate (PER) upper limit (TRM/CA/12/C)	:CONFigure:SRWireless:BT:EPER:ULIMit/?
Set/query the BT EDR in-band spurious emissions upper limits (TRM/CA/13/C)	:CONFigure:SRWireless:BT:IBSPurious:ULIMit/?
Set/query the BT BR 20 dB bandwidth upper limit (TRM/CA/05/C)	:CONFigure:SRWireless:BT:BANDwidth:ULIMit/?

***BLUETOOTH* LOW ENERGY**

Function	Command / Query
Set/query BLE average output power lower limit (TRM-LE/CA/BV-01-C)	:CONFigure:SRWireless:BLE:APOWer:LLIMit/?
Set/query BLE average output power upper limit (TRM-LE/CA/BV-01-C)	:CONFigure:SRWireless:BLE:APOWer:ULIMit/?
Set/query BLE crest factor upper limit (TRM-LE/CA/BV-01-C)	:CONFigure:SRWireless:BLE:CFACtor:ULIMit/?
Set/query BLE (1 Mbps) in-band emissions upper limits (TRM-LE/CA/BV-03-C)	:CONFigure:SRWireless:BLE:IBEMissions:ULIMit/?
Set/query 2LE (BLE 2 Mbps) in-band emissions upper limits (TRM-LE/CA/BV-08-C)	:CONFigure:SRWireless:BLE:B2LE:IBEMissions:ULIMit/?
Set/query BLE (1 Mbps)Δ f1 average lower limit (TRM-LE/CA/BV-05-C, TRM-LE/CA/BV-09-C)	:CONFigure:SRWireless:BLE:F1AVerage:LLIMit/?
Set/query 2LE (BLE 2 Mbps) Δf1 average lower limit (TRM-LE/CA/BV-10-C, TRM-LE/CA/BV-11-C)	:CONFigure:SRWireless:BLE:B2LE:F1AVerage:LLIMit/?
Set/query BLR (BLE coded S=8) Δf1 average lower limit (TRM-LE/CA/BV-13-C)	:CONFigure:SRWireless:BLE:BLR:F1AVerage:LLIMit/?
Set/query BLE (1 Mbps)Δ f1 average upper limit (TRM-LE/CA/BV-05-C, TRM-LE/CA/BV-09-C)	:CONFigure:SRWireless:BLE:F1AVerage:ULIMit/?
Set/query 2LE (BLE 2 Mbps) Δf1 average upper limit (TRM-LE/CA/BV-10-C, TRM-LE/CA/BV-11-C)	:CONFigure:SRWireless:BLE:B2LE:F1AVerage:ULIMit/?
Set/query BLR (BLE coded S=8) Δf1 average upper limit (TRM-LE/CA/BV-13-C)	:CONFigure:SRWireless:BLE:BLR:F1AVerage:ULIMit/?
Set/query BLE (1 Mbps)Δ f2 maximum lower limit (TRM-LE/CA/BV-05-C, TRM-LE/CA/BV-09-C)	:CONFigure:SRWireless:BLE:F2Maximum:LLIMit/?
Set/query 2LE (BLE 2 Mbps) Δf2 maximum lower limit (TRM-LE/CA/BV-10-C,	:CONFigure:SRWireless:BLE:B2LE:F2Maximum:LLI

Function	Command / Query
TRM-LE/CA/BV-11-C)	Mit/?
Set/query BLR (BLE coded S=8) Δf1 maximum lower limit (TRM-LE/CA/BV-13-C)	:CONFigure:SRWireless:BLE:BLR:F1Maximum:LLIMit/?
Set/query BLE (1 Mbps)Δ f2 average / f1 average lower limit (TRM-LE/CA/BV-05-C, TRM-LE/CA/BV-09-C)	:CONFigure:SRWireless:BLE:F21Ratio:LLIMit/?
Set/query 2LE (BLE 2 Mbps) Δf2 average / Δf1 average lower limit (TRM-LE/CA/BV-10-C, TRM-LE/CA/BV-11-C)	:CONFigure:SRWireless:BLE:B2LE:F21Ratio:LLIMit/?
Set/query BLE (1 Mbps and 2 Mbps) carrier frequency offset limit (TRM-LE/BV-06-C and TRM-LE/CA/BV-12-C)	:CONFigure:SRWireless:BLE:CFOFfset:LIMit/?
Set/query BLR (BLE coded S=8) carrier frequency offset limit (TRM-LE/CA/BV-14-C)	:CONFigure:SRWireless:BLE:BLR:CFOFfset:LIMit/?
Set/query BLE (1 Mbps and 2 Mbps) carrier frequency drift (TRM-LE/CA/BV-06-C and TRM-LE/CA/BV-12-C)	:CONFigure:SRWireless:BLE:CDRift:LIMit/?
Set/query BLR (BLE coded S=8) carrier frequency drift limit (TRM-LE/CA/BV-14-C)	:CONFigure:SRWireless:BLE:BLR:CDRift:LIMit/?
Set/query BLE (1 Mbps and 2 Mbps) carrier frequency drift rate limit (TRM-LE/CA/BV-06-C and TRM-LE/CA/BV-12-C)	:CONFigure:SRWireless:BLE:DRATe:LIMit/?
Set/query BLR (BLE coded S=8) carrier frequency drift rate limit (TRM-LE/CA/BV-14-C)	:CONFigure:SRWireless:BLE:BLR:DRATe:LIMit/?
Set/query BLE (1 Mbps and 2 Mbps) initial carrier frequency drift rate limit (TRM-LE/CA/BV-06-C and TRM-LE/CA/BV-12-C)	:CONFigure:SRWireless:BLE:IDRate:LIMit/?
Set/query BLR (BLE coded S=8) initial carrier frequency drift rate limit (TRM-LE/CA/BV-14-C)	:CONFigure:SRWireless:BLE:BLR:IDRate:LIMit/?

5.2.8 Active Segment Measurements

Function	Command / Query
Sets the instrument input port to be used for the active segment.	:CONFigure:SRWireless:SEGMent:PORT
Enable/Disable the active segment.	:CONFigure:SRWireless:SEGMent:STATe

Non-Packetised: CW and Continuously Modulated

Function	Command / Query
Enable/disable power (average and peak)	:CONFigure:SRWireless:SELect:CW:POWer/?
Enable/disable frequency	:CONFigure:SRWireless:SELect:CW:FREQuency/?
Enable/disable spectrum	:CONFigure:SRWireless:SELect:CW:SPECtrum/?

WLAN - NUMERIC

Function	Command / Query
Enable/disable transmit power (numeric)	:CONFigure:SRWireless:SELect:WLAN:POWer/?
Enable/disable power profile (numeric) - power on and power down ramps – DSSS packets	:CONFigure:SRWireless:SELect:WLAN:DSSS:PRAMps/?
Enable / disable power on and power down ramp measurements on WLAN packets (both DSSS and OFDM).	:CONFigure:SRWireless:SELect:WLAN:PRAMps
Enable/disable EVM (basic - numeric)	:CONFigure:SRWireless:SELect:WLAN:EVM[:BASic]/?
Enable / disable WLAN combined EVM measurements.	:CONFigure:SRWireless:SELect:WLAN:EVM:COMBined/?
Enable/disable transmitter spectrum (numeric)	:CONFigure:SRWireless:SELect:WLAN:SPECtrum:NUMeric/?
Enable/disable composite MIMO spatial stream power.	:CONFigure:SRWireless:SELect:WLAN:SPOWer
Enable/disable sequential and true MIMO cross power measurements	:CONFigure:SRWireless:SELect:WLAN:CPOWer
Enable/disable the availability of channel estimation values for OFDM signals.	:CONFigure:SRWireless:SELect:WLAN:CESTimation

WLAN - GRAPHICAL

Function	Command / Query
Enable/disable power profile (graphical)	:CONFigure:SRWireless:SELect:WLAN:PPRofile/?
Enable/disable CCDF profile (graphical)	:CONFigure:SRWireless:SELect:WLAN:CCDF/?
Enable/disable constellation (graphical)	:CONFigure:SRWireless:SELect:WLAN:CONSt/?
Enable/disable transmitter spectrum (graphical)	:CONFigure:SRWireless:SELect:WLAN:SPECtrum:GRAPh/?
Enable/disable OFDM preamble frequency error v time measurements	:CONFigure:SRWireless:SELect:WLAN:OFDM:FERRor/?
Enable/disable OFDM phase error v symbol measurements	:CONFigure:SRWireless:SELect:WLAN:OFDM:PERRor/?

BLUETOOTH - NUMERIC

Function	Command / Query
Enable/disable *Bluetooth* transmit power (TRM/CA/01/C)	:CONFigure:SRWireless:SELect:BT:POWer/?
Enable/disable *Bluetooth* frequency range (TRM/CA/04/C)	:CONFigure:SRWireless:SELect:BT:FRANge/?
Enable/disable *Bluetooth* 20 dB bandwidth (TRM/CA/05/C)	:CONFigure:SRWireless:SELect:BT:BANDwidth/?
Enable/disable *Bluetooth* initial carrier frequency tolerance (TRM/CA/08/C)	:CONFigure:SRWireless:SELect:BT:ICFTolerance/?
Enable/disable *Bluetooth* carrier drift (TRM/CA/09/C)	:CONFigure:SRWireless:SELect:BT:CDRift/?
Enable/disable *Bluetooth* modulation characteristics (TRM/CA/07/C)	:CONFigure:SRWireless:SELect:BT:MODulation/?
Enable / disable *Bluetooth* measurements – adjacent channel power (TRM/CA/06/C)	:CONFigure:SRWireless:SELect:BT:ACPower
Enable/disable *Bluetooth* EDR relative TX power (TRM/CA/10/C)	:CONFigure:SRWireless:SELect:BT:ERTPower/?
Enable/disable *Bluetooth* EDR carrier freq. stability and mod. accuracy (TRM/CA/11/C)	:CONFigure:SRWireless:SELect:BT:ECModulation/?
Enable/disable *Bluetooth* EDR differential phase encoding (TRM/CA/12/C)	:CONFigure:SRWireless:SELect:BT:EDPencoding/?
Enable/disable *Bluetooth* EDR in-band spurious emissions (TRM/CA/13/C)	:CONFigure:SRWireless:SELect:BT:IBSPurious?
Enable/disable *Bluetooth* LE output power (TRM-LE/CA/BV-01-C)	:CONFigure:SRWireless:SELect:BLE:POWer/?
Enable/disable *Bluetooth* LE in-band emissions measurement(TRM-LE/CA/BV-03-C, TRM-LE/CA/BV-08-C)	:CONFigure:SRWireless:SELect:BLE:IBEMissions/?
Enable/disable *Bluetooth* LE modulation characteristics (TRM-LE/CA/BV-05-C, TRM-LE/CA/BV-09-C, TRM-LE/CA/BV-10-C, TRM-LE/CA/BV-11-C, and TRM-LE/CA/BV-13-C)	:CONFigure:SRWireless:SELect:BLE:MODulation/?
Enable/disable *Bluetooth* LE carrier frequency offset and drift (TRM-LE/CA/BV-06-C, TRM-LE/CA/BV-12-C and TRM-LE/CA/BV-14-C)	:CONFigure:SRWireless:SELect:BLE:CDRift/?

BLUETOOTH – GRAPHICAL

Function	Command / Query
Enable/disable *Bluetooth* power burst profile (BR / EDR / BLE)	:CONFigure:SRWireless:SELect:BT:PPRofile/?
Enable/disable *Bluetooth* deviation vs symbol (BR / BLE)	:CONFigure:SRWireless:SELect:BT:DEViation/?
Enable/disable *Bluetooth* eye diagram (BR / BLE)	:CONFigure:SRWireless:SELect:BT:EYE/?
Enable/disable *Bluetooth* spectrum (BR / EDR / BLE)	:CONFigure:SRWireless:SELect:BT:SPECtrum/?
Enable/disable *Bluetooth* vector diagram (EDR)	:CONFigure:SRWireless:SELect:BT:VECTor/?
Enable/disable *Bluetooth* constellation (EDR)	:CONFigure:SRWireless:SELect:BT:CONSt/?

Function	Command / Query
Enable/disable *Bluetooth* DEVM vs symbol (EDR)	:CONFigure:SRWireless:SELect:BT:DEVM/?

5.2.9 Active Segment Gates

Function	Command / Query
Add a gate to WLAN transmit power measurement	:CONFigure:SRWireless:WLAN:POWer:GATE:ADD
Remove a gate from WLAN transmit power measurement	:CONFigure:SRWireless:WLAN:POWer:GATE:REMove
Query gates assigned to WLAN transmit power measurement	:CONFigure:SRWireless:WLAN:POWer:GATE[:STATus]?
Assign a gate to WLAN spectrum measurement	:CONFigure:SRWireless:WLAN:SPECtrum:GATE:ADD
Remove a gate from WLAN spectrum measurement	:CONFigure:SRWireless:WLAN:SPECtrum:GATE:REMove
Query gates assigned to WLAN spectrum measurement.	:CONFigure:SRWireless:WLAN:SPECtrum:GATE[:STATus]?
Assign a gate to WLAN CCDF measurement	:CONFigure:SRWireless:WLAN:CCDF:GATE:ADD
Remove a gate from WLAN CCDF measurement	:CONFigure:SRWireless:WLAN:CCDF:GATE:REMove
Query gates assigned to WLAN CCDF measurement	:CONFigure:SRWireless:WLAN:CCDF:GATE[:STATus]?

5.2.10 Capture

Function	Command / Query
Initiate capture and analysis	:INITiate:SRWireless
Abort capture and analysis	:ABORt:SRWireless
Perform analysis on existing captured data	:CALC:SRWireless
Query measurement status	:STATus:SRWireless:MEASurement?
Check the status of a measurement in progress	:STATus:SRWireless:MEASurement?
Set the capture time used during each stage of the automatic leveling process.	:CONFigure:SRWireless:ALEVel:TIME
Initiate the automatic leveling function.	:INITiate:SRWireless:ALEVel
Set the trigger timeout for use when the trigger mode is set to Level.	:CONFigure:SRWireless:TTIMeout

Function	Command / Query
Set or query the capture mode for the active segment.	:CONFigure:SRWireless:CAPTure:MODE
Adjust the minimum inter-packet gap recognised by the packet detector.	:CONFigure:SRWireless:PDETector:MGAP
Set/query the minimum length of packet that will be recognized by the packet detector.	:CONFigure:SRWireless:PDETector:MLENgth/?
Set/query the state of the function to synchronize to the next packet following settling time. This function is used to avoid analyzing packet fragments at the start of captured data.	:CONFigure:SRWireless:STPSync/?

5.2.11 Capture Information

Function	Command / Query
Query capture information (segment and packet edge locations and meas valid status)	:FETCh:SRWireless:CINFormation?
Query capture information (Extended version requiring extension code)	:FETCh:SRWireless:CINFormation? (Extended)
Fetch full capture information, including that of "hidden" segments.	:FETCh:SRWireless:IQ:INFOrmation?
Query capture profile data from a specified portion of the capture - binary format	:FETCh:SRWireless:CPRofile?
Query IQ data from a specified portion of the capture. - binary format	:FETCh:SRWireless:IQ?
Save the captured IQ data in a file on the instrument in binary format for later retrieval using FTP.	:CALCulate:SRWireless:IQ:SAVE
Fetch information about individual packets that have been captured and identified. automatically	:FETCh:SRWireless:PACKet:IDENtity?
Fetch information about packets that have been captured and identified automatically.	:FETCh:SRWireless:SEGMent:IDENtity?
Write information about a capture to the instrument	:CONFigure:SRWireless:IQ:INFormation:WRITe
Upload binary IQ data to the instrument	:CONFigure:SRWireless:IQ:LOAD[:BINary]

5.2.12 Fetch Summary Measurements for Segment

Non-Packetised: CW and Continuously Modulated

Function	Command / Query
Fetch CW power (average and peak)	:FETCh:SRWireless:SUMMary:CW:POWer?
Fetch CW frequency	:FETCh:SRWireless:SUMMary:CW:FREQuency?

Function	Command / Query
Fetch CW frequency with optional extension code	:FETCh:SRWireless:SUMMary:CW:FREQuency? (Extended)
Fetch CW spectrum – binary format	:FETCh:SRWireless:SUMMary:CW:SPECtrum[:BINary]?
Fetch CW spectrum – ASCII format	:FETCh:SRWireless:SUMMary:CW:SPECtrum:ASCii?

WLAN - NUMERIC

Function	Command / Query
Fetch transmit power (numeric) measurements.	:FETCh:SRWireless:SUMMary:WLAN:POWer?
Fetch power profile (numeric) measurements - power up and down ramps	:FETCh:SRWireless:SUMMary:WLAN:DSSS:PRAMps?
Fetch EVM (basic - numeric) measurements - DSSS packets	:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM?
Fetch basic WLAN EVM measurements - DSSS packets - using extension code	:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM? (Extended)
Fetch WLAN EVM measurements in units of percent.	:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:PCT?
Fetch WLAN EVM measurements in dB units.	:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:DB?
Fetch EVM (basic - numeric) measurements - OFDM packets	:FETCh:SRWireless:SUMMary:WLAN:OFDM:EVM?
Fetch EVM (basic - numeric) measurements - OFDM packets - using extension code	:FETCh:SRWireless:SUMMary:WLAN:OFDM:EVM? (Extended)
Fetch combined WLAN EVM measurements.	:FETCh:SRWireless:SUMMary:WLAN:OFDM:EVM:COMBined?
Fetch EVM (advanced - numeric) measurements	:FETCh:SRWireless:SUMMary:WLAN:OFDM:AEVM?
Fetch the channel estimation values (amplitude and phase)	:FETCh:SRWireless:SUMMary:WLAN:OFDM:CESTimation?
Fetch the channel estimation values (amplitude and phase)	:FETCh:SRWireless:SUMMary:WLAN:OFDM:CESTimation:AMPLitude?
Fetch the channel estimation values (amplitude and phase)	:FETCh:SRWireless:SUMMary:WLAN:OFDM:CESTimation:PHASe?
Fetch transmitter spectrum (numeric) measurements	:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:NUMeric?
Fetch WLAN transmitter spectrum (numeric) measurements with optional extension code for returning measurements for each segment of the spectral mask.	:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:NUMeric? (Extended)
Fetch spot spectrum measurements at a frequency offset above and below the carrier frequency (fc).	:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:SPOT?

Function	Command / Query
Fetch WLAN power profile (numeric) measurements - power up and down ramps – WLAN packets	:FETCh:SRWireless:SUMMary:WLAN:PRAMps?
Fetch CCDF power distribution spot measurement, corresponding to the ‘percentage of samples’ value set using CONF:SRW:WLAN:CCDF:PSAM.	:FETCh:SRWireless:SUMMary:WLAN:CCDF:PDISt?
Fetch composite MIMO stream power and related measurements.	:FETCh:SRWireless:SUMMary:WLAN:SPOWer?
Fetch true or sequential MIMO cross power measurements	:FETCh:SRWireless:SUMMary:WLAN:CPOWer?

WLAN - GRAPHICAL

Function	Command / Query
Fetch power profile (graphical) measurements – average – ASCII format	:FETCh:SRWireless:SUMMary:WLAN:PPRofile:AVERage:ASCii?
Fetch power profile (graphical) measurements – average – binary format	:FETCh:SRWireless:SUMMary:WLAN:PPRofile:AVERage[:BINary]?
Fetch power profile (graphical) measurements – peak – ASCII format	:FETCh:SRWireless:SUMMary:WLAN:PPRofile:PEAK:ASCii?
Fetch power profile (graphical) measurements – peak – binary format	:FETCh:SRWireless:SUMMary:WLAN:PPRofile:PEAK[:BINary]?
Fetch CCDF profile (graphical) measurements - ASCII format	:FETCh:SRWireless:SUMMary:WLAN:CCDF:ASCii?
Fetch CCDF profile (graphical) measurements - binary format	:FETCh:SRWireless:SUMMary:WLAN:CCDF[:BINary]?
Fetch transmitter spectrum (graphical) measurements - ASCII format	:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:GRAPh:ASCii?
Fetch transmitter spectrum (graphical) measurements - binary format	:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:GRAPh[:BINary]?
Fetch WLAN transmitter spectrum (graphical) combined 802.11ac 80+80 measurements - ASCII format.	:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:GRAPh:COMBined:ASCii?
Fetch WLAN transmitter spectrum (graphical) combined 802.11ac 80+80 measurements - binary format.	:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:GRAPh:COMBined[:BINary]?
Fetch WLAN transmitter spectrum mask for 802.11ac 80+80 measurements - ASCII format.	:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:MASK:COMBined:ASCii?
Fetch WLAN transmitter spectrum mask for 802.11ac 80+80 measurements - binary format.	:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:MASK:COMBined[:BINary]?
Fetch OFDM preamble frequency error v time measurement	:FETCh:SRWireless:SUMMary:WLAN:OFDM:FERRor?
Fetch OFDM phase error v symbol measurement	:FETCh:SRWireless:SUMMary:WLAN:OFDM:PERRor?

BLUETOOTH - NUMERIC

Function	Command / Query
Fetch *Bluetooth* transmit power (TRM/CA/01/C)	:FETCh:SRWireless:SUMMary:BT:POWer?
Fetch *Bluetooth* frequency range (TRM/CA/04/C)	:FETCh:SRWireless:SUMMary:BT:FRANge?
Fetch *Bluetooth* 20 dB bandwidth (TRM/CA/05/C)	:FETCh:SRWireless:SUMMary:BT:BANDwidth?
Fetch *Bluetooth* adjacent channel power (TRM/CA/06/C).	:FETCh:SRWireless:SUMMary:BT:ACPower?
Fetch *Bluetooth* initial carrier frequency tolerance (TRM/CA/08/C)	:FETCh:SRWireless:SUMMary:BT:ICFTolerance?
Fetch *Bluetooth* carrier drift (TRM/CA/09/C)	:FETCh:SRWireless:SUMMary:BT:CDRift?
Fetch *Bluetooth* modulation characteristics (TRM/CA/07/C)	:FETCh:SRWireless:SUMMary:BT:MODulation?
Fetch *Bluetooth* EDR relative TX power (TRM/CA/10/C)	:FETCh:SRWireless:SUMMary:BT:ERTPower?
Fetch *Bluetooth* EDR carrier freq. stability and mod. accuracy (TRM/CA/11/C)	:FETCh:SRWireless:SUMMary:BT:ECModulation?
Fetch *Bluetooth* EDR carrier freq. stability and mod. accuracy (TRM/CA/11/C)	:FETCh:SRWireless:SUMMary:BT:ECModulation? (Extended)
Fetch Bluetooth EDR differential phase encoding (TRM/CA/12/C)	:FETCh:SRWireless:SUMMary:BT:EDPencoding?
Fetch the *Bluetooth* EDR in-band spurious emissions (TRM/CA/13/C)	:FETCh:SRWireless:SUMMary:BT:IBSPurious?
Fetch *Bluetooth* LE output power (TRM-LE/CA/BV-01-C)	:FETCh:SRWireless:SUMMary:BLE:POWer?
Fetch *Bluetooth* LE in-band emissions (TRM-LE/CA/BV-03-C and TRM-LE/CA/BV-08-C)	:FETCh:SRWireless:SUMMary:BLE:IBEMissions?
Fetch *Bluetooth* LE modulation characteristics (TRM-LE/CA/BV-05-C, TRM-LE/CA/BV-09-C, TRM-LE/CA/BV-10-C, TRM-LE/CA/BV-11-C, and TRM-LE/CA/BV-13-C)	:FETCh:SRWireless:SUMMary:BLE:MODulation?
Fetch *Bluetooth* LE carrier frequency offset and drift (TRM-LE/CA/BV-06-C, TRM-LE/CA/BV-12-C and TRM-LE/CA/BV-14-C)	:FETCh:SRWireless:SUMMary:BLE:CDRift?
Fetch *Bluetooth* LE carrier frequency offset and drift (TRM-LE/CA/BV-06-C, TRM-LE/CA/BV-12-C and TRM-LE/CA/BV-14-C). An optional extension code returns measurements in addition to those required by the standard.	:FETCh:SRWireless:SUMMary:BLE:CDRift? (Extended)

BLUETOOTH - GRAPHICAL

Function	Command / Query
Fetch *Bluetooth* power burst profile (BR / EDR) – averaged data – ASCII	:FETCh:SRWireless:SUMMary:BT:PPRofile:AVERage:ASCi i?
Fetch *Bluetooth* power burst profile (BR / EDR) – averaged data – binary	:FETCh:SRWireless:SUMMary:BT:PPRofile:AVERage[:BIN ary]?

Function	Command / Query
Fetch *Bluetooth* power burst profile (BR / EDR) – peak data – ASCii	:FETCh:SRWireless:SUMMary:BT:PPRofile:PEAK:ASCii?
Fetch *Bluetooth* power burst profile (BR / EDR) – peak data – binary	:FETCh:SRWireless:SUMMary:BT:PPRofile:PEAK[:BINary]?
Fetch *Bluetooth* deviation vs symbol (BR) – averaged data – ASCii	:FETCh:SRWireless:SUMMary:BT:DEViation:ASCii?
Fetch *Bluetooth* deviation vs symbol (BR) – averaged data – binary	:FETCh:SRWireless:SUMMary:BT:DEViation[:BINary]?
Fetch *Bluetooth* power burst profile (BLE) – averaged data – ASCII	:FETCh:SRWireless:SUMMary:BLE:PPRofile:AVERage:ASCii?
Fetch *Bluetooth* power burst profile (BLE) – averaged data – binary	:FETCh:SRWireless:SUMMary:BLE:PPRofile:AVERage[:BINary]?
Fetch *Bluetooth* power burst profile (BLE) – peak data – ASCii	:FETCh:SRWireless:SUMMary:BLE:PPRofile:PEAK:ASCii?
Fetch *Bluetooth* power burst profile (BLE) – peak data – binary	:FETCh:SRWireless:SUMMary:BLE:PPRofile:PEAK[:BINary]?
Fetch *Bluetooth* deviation vs symbol (BLE) – averaged data – ASCii	:FETCh:SRWireless:SUMMary:BLE:DEViation:ASCii?
Fetch *Bluetooth* deviation vs symbol (BLE) – averaged data – binary	:FETCh:SRWireless:SUMMary:BLE:DEViation[:BINary]?
Fetch *Bluetooth* eye diagram (BR / BLE) – ascii format	:FETCh:SRWireless:SUMMary:BT:EYE:ASCii?
Fetch *Bluetooth* eye diagram (BR / BLE) – binary format	:FETCh:SRWireless:SUMMary:BT:EYE[:BINary]?
Fetch *Bluetooth* spectrum (BR / EDR / BLE) – ascii format	:FETCh:SRWireless:SUMMary:BT:SPECtrum:ASCii?
Fetch *Bluetooth* spectrum (BR / EDR / BLE) – binary format	:FETCh:SRWireless:SUMMary:BT:SPECtrum[:BINary]?
Fetch *Bluetooth* DEVM vs symbol (EDR) – ascii format	:FETCh:SRWireless:SUMMary:BT:DEVM:ASCii?
Fetch *Bluetooth* DEVM vs symbol (EDR) – binary format	:FETCh:SRWireless:SUMMary:BT:DEVM[:BINary]?

5.2.13 Fetch Measurements for Packet

WLAN - Numeric

Function	Command / Query
Fetch transmit power (numeric) measurements.	:FETCh:SRWireless:PACKet:WLAN:POWer?
Fetch power profile (numeric) measurements - power up and down ramps	:FETCh:SRWireless:PACKet:WLAN:DSSS:PRAMps?
Fetch EVM (basic - numeric) measurements - DSSS packets	:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM?
Fetch EVM (basic - numeric) measurements - DSSS packets – using extension code	:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM? (Extended)
Fetch WLAN EVM measurements in units of percent.	:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:PCT?
Fetch WLAN EVM measurements in dB units.	:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:DB?

Function	Command / Query
Fetch EVM (basic - numeric) measurements - OFDM packets	:FETCh:SRWireless:PACKet:WLAN:OFDM:EVM?
Fetch EVM (basic - numeric) measurements - OFDM packets - using extension code	:FETCh:SRWireless:PACKet:WLAN:OFDM:EVM? (Extended)
Fetch combined WLAN EVM measurements.	:FETCh:SRWireless:PACKet:WLAN:OFDM:EVM:COMBined?
Fetch EVM (advanced - numeric) measurements	:FETCh:SRWireless:PACKet:WLAN:OFDM:AEVM?
Fetch the channel estimation values (amplitude and phase)	:FETCh:SRWireless:PACKet:WLAN:OFDM:CESTimation?
Fetch the channel estimation values (amplitude and phase)	:FETCh:SRWireless:PACKet:WLAN:OFDM:CESTimation:AMPLitude?
Fetch the channel estimation values (amplitude and phase)	:FETCh:SRWireless:PACKet:WLAN:OFDM:CESTimation:PHASe?
Fetch transmitter spectrum (numeric) measurements	:FETCh:SRWireless:PACKet:WLAN:SPECtrum:NUMeric?
Fetch WLAN transmitter spectrum (numeric) measurements with optional extension code for returning measurements for each segment of the spectral mask.	:FETCh:SRWireless:PACKet:WLAN:SPECtrum:NUMeric? (Extended)
Fetch spot spectrum measurements at a frequency offset above and below the carrier frequency (fc).	:FETCh:SRWireless:PACKet:WLAN:SPECtrum:SPOT?
Fetch WLAN power profile (numeric) measurements - power up and down ramps – WLAN packets.	: FETCh:SRWireless:PACKet:WLAN:PRAMps?
Fetch CCDF power distribution spot measurement, corresponding to the 'percentage of samples' value set using CONF:SRW:WLAN:CCDF:PSAM.	:FETCh:SRWireless:PACKet:WLAN:CCDF:PDISt?
Fetch composite MIMO stream power and related measurements.	:FETCh:SRWireless:PACKet:WLAN:SPOWer?
Fetch true or sequential MIMO cross power measurements	:FETCh:SRWireless:PACKet:WLAN:CPOWer?

WLAN - GRAPHICAL

Function	Command / Query
Fetch power profile (graphical) measurements – average – ASCII format	:FETCh:SRWireless:PACKet:WLAN:PPRofile:AVERage:ASCii?
Fetch power profile (graphical) measurements – average – binary format	:FETCh:SRWireless:PACKet:WLAN:PPRofile:AVERage[:BINary]?
Fetch power profile (graphical) measurements – peak – ASCII format	:FETCh:SRWireless:PACKet:WLAN:PPRofile:PEAK:ASCii?
Fetch power profile (graphical) measurements – peak – binary format	:FETCh:SRWireless:PACKet:WLAN:PPRofile:PEAK[:BINary]?
Fetch CCDF profile (graphical) measurements - ASCII format	:FETCh:SRWireless:PACKet:WLAN:CCDF:ASCii?

Function	Command / Query
Fetch CCDF profile (graphical) measurements - binary format	:FETCh:SRWireless:PACKet:WLAN:CCDF[:BINary]?
Fetch constellation (graphical) measurements – OFDM packets – binary format	:FETCh:SRWireless:PACKet:WLAN:OFDM:CONSt[:BINary]?
Fetch constellation (graphical) measurements – DSSS packets – binary format	:FETCh:SRWireless:PACKet:WLAN:DSSS:CONSt[:BINary]?
Fetch transmitter spectrum (graphical) measurements - ASCII format	:FETCh:SRWireless:PACKet:WLAN:SPECtrum:GRAPh:ASCii?
Fetch transmitter spectrum (graphical) measurements - binary format	:FETCh:SRWireless:PACKet:WLAN:SPECtrum:GRAPh[:BINary]?
Fetch WLAN transmitter spectrum (graphical) combined 802.11ac 80+80 measurements - ASCII format.	:FETCh:SRWireless:PACKet:WLAN:SPECtrum:GRAPh:COMBined:ASCii?
Fetch WLAN transmitter spectrum (graphical) combined 802.11ac 80+80 measurements - binary format.	:FETCh:SRWireless:PACKet:WLAN:SPECtrum:GRAPh:COMBined[:BINary]?
Fetch WLAN transmitter spectrum mask for 802.11ac 80+80 measurements - ASCII format.	:FETCh:SRWireless:PACKet:WLAN:SPECtrum:MASK:COMBined:ASCii?
Fetch WLAN transmitter spectrum mask for 802.11ac 80+80 measurements - binary format.	:FETCh:SRWireless:PACKet:WLAN:SPECtrum:MASK:COMBined[:BINary]?
Fetch OFDM preamble frequency error v time measurement	:FETCh:SRWireless:PACKet:WLAN:OFDM:FERRor?
Fetch OFDM phase error v symbol measurement	:FETCh:SRWireless:PACKet:WLAN:OFDM:PERRor?

BLUETOOTH - NUMERIC

Function	Command / Query
Fetch *Bluetooth* transmit power (TRM/CA/01/C)	:FETCh:SRWireless:PACKet:BT:POWer?
Fetch *Bluetooth* frequency range (TRM/CA/04/C)	:FETCh:SRWireless:PACKet:BT:FRANge?
Fetch *Bluetooth* 20 dB bandwidth (TRM/CA/05/C)	:FETCh:SRWireless:PACKet:BT:BANDwidth?
Fetch *Bluetooth* adjacent channel power (TRM/CA/06/C)	:FETCh:SRWireless:PACKet:BT:ACPower?
Fetch *Bluetooth* initial carrier frequency tolerance (TRM/CA/08/C)	:FETCh:SRWireless:PACKet:BT:ICFTolerance?
Fetch *Bluetooth* carrier drift (TRM/CA/09/C)	:FETCh:SRWireless:PACKet:BT:CDRift?
Fetch *Bluetooth* modulation characteristics (TRM/CA/07/C)	:FETCh:SRWireless:PACKet:BT:MODulation?
Fetch *Bluetooth* EDR relative TX power (TRM/CA/10/C)	:FETCh:SRWireless:PACKet:BT:ERTPower?
Fetch *Bluetooth* EDR carrier frequency stability and modulation accuracy (TRM/CA/11/C)	:FETCh:SRWireless:PACKet:BT:ECModulation?
Fetch *Bluetooth* EDR carrier frequency stability and modulation accuracy	:FETCh:SRWireless:PACKet:BT:ECModulation? (Extended)

Function	Command / Query
(TRM/CA/11/C)	
Fetch *Bluetooth* EDR diffential phase encoding (TRM/CA/12/C)	:FETCh:SRWireless:PACKet:BT:EDPencoding?
Fetch *Bluetooth* EDR in-band spurious emissions (TRM/CA/13/C)	:FETCh:SRWireless:PACKet:BT:IBSPurious?
Fetch *Bluetooth* LE output power (TRM-LE/CA/BV-01-C)	:FETCh:SRWireless:PACKet:BLE:POWer?
Fetch *Bluetooth* LE in-band spurious emissions (TRM-LE/CA/BV-03-C and TRM-LE/CA/BV-08-C)	:FETCh:SRWireless:PACKet:BLE:IBEMissions?
Fetch *Bluetooth* LE modulation characteristics (TRM-LE/CA/BV-05-C, TRM-LE/CA/BV-09-C, TRM-LE/CA/BV-10-C, TRM-LE/CA/BV-11-C, and TRM-LE/CA/BV-13-C)	:FETCh:SRWireless:PACKet:BLE:MODulation?
Fetch *Bluetooth* LE carrier frequency offset and drift (TRM-LE/CA/BV-06-C, TRM-LE/CA/BV-12-C and TRM-LE/CA/BV-14-C)	:FETCh:SRWireless:PACKet:BLE:CDRift?
Fetch *Bluetooth* LE carrier frequency offset and drift (TRM-LE/CA/BV-06-C, TRM-LE/CA/BV-12-C and TRM-LE/CA/BV-14-C). An optional extension code returns measurements in addition to those required by the standard.	:FETCh:SRWireless:PACKet:BLE:CDRift? (Extended)

BLUETOOTH - GRAPHICAL

Function	Command / Query
Fetch *Bluetooth* power burst profile (BR / EDR) - averaged data - ASCII	:FETCh:SRWireless:PACKet:BT:PPRofile:AVERage:ASCii?
Fetch *Bluetooth* power burst profile (BR / EDR) - averaged data - binary	:FETCh:SRWireless:PACKet:BT:PPRofile:AVERage[:BINary]?
Fetch *Bluetooth* power burst profile (BR / EDR) - peak data - ASCII	:FETCh:SRWireless:PACKet:BT:PPRofile:PEAK:ASCii?
Fetch *Bluetooth* power burst profile (BR / EDR) - peak data - binary	:FETCh:SRWireless:PACKet:BT:PPRofile:PEAK[:BINary]?
Fetch *Bluetooth* deviation vs symbol (BR) - ASCII	:FETCh:SRWireless:PACKet:BT:DEViation:ASCii?
Fetch *Bluetooth* deviation vs symbol (BR) - binary	:FETCh:SRWireless:PACKet:BT:DEViation[:BINary]?
Fetch *Bluetooth* power burst profile (BLE) - averaged data - ASCII	:FETCh:SRWireless:PACKet:BLE:PPRofile:AVERage:ASCii?
Fetch *Bluetooth* power burst profile (BLE) - averaged data - binary	:FETCh:SRWireless:PACKet:BLE:PPRofile:AVERage[:BINary]?
Fetch *Bluetooth* power burst profile (BLE) - peak data - ASCII	:FETCh:SRWireless:PACKet:BLE:PPRofile:PEAK:ASCii?
Fetch *Bluetooth* power burst profile (BLE) - peak data - binary	:FETCh:SRWireless:PACKet:BLE:PPRofile:PEAK[:BINary]?
Fetch *Bluetooth* deviation vs symbol (BLE) - ASCII	:FETCh:SRWireless:PACKet:BLE:DEViation:ASCii?
Fetch *Bluetooth* deviation vs symbol (BLE) - binary	:FETCh:SRWireless:PACKet:BLE:DEViation[:BINary]?

Function	Command / Query
Fetch *Bluetooth* eye diagram (BR / BLE) - ASCII	:FETCh:SRWireless:PACKet:BT:EYE:ASCii?
Fetch *Bluetooth* eye diagram (BR / BLE) - binary	:FETCh:SRWireless:PACKet:BT:EYE[:BINary]?
Fetch *Bluetooth* spectrum (BR / EDR / BLE) - ASCii	:FETCh:SRWireless:PACKet:BT:SPECtrum:ASCii?
Fetch *Bluetooth* spectrum (BR / EDR / BLE) - binary	:FETCh:SRWireless:PACKet:BT:SPECtrum[:BINary]?
Fetch *Bluetooth* vector diagram (EDR)	:FETCh:SRWireless:PACKet:BT:VECTor[:BINary]?
Fetch *Bluetooth* constellation (EDR)	:FETCh:SRWireless:PACKet:BT:CONSt[:BINary]?
Fetch *Bluetooth* DEVM vs symbol (EDR) - ASCII	:FETCh:SRWireless:PACKet:BT:DEVM:ASCii?
Fetch *Bluetooth* DEVM vs symbol (EDR) - binary	:FETCh:SRWireless:PACKet:BT:DEVM[:BINary]?

5.3 SCPI Data Formats

5.3.1 Program Data

Format	Description	Examples

Selection	Known as Character Program Data and Character Response data in IEEE-488.2, this format is used to represent system states that are best described by a string of characters.	`USER`
		`AUTO`
		`MCS7`

Number	Known as Decimal Numeric Program Data in IEEE-488.2, a *Number* can be an integer or a real number with or without an exponent field. The value may be followed by a suffix comprising a unit preceded by an optional multiplier.	`-123`
		`-12.34`
		`-1.234E+01`
		`12.34 dBm`
		`2402 MHz`
		`10 us`

Number+	This corresponds to the Numeric Value format defined in the SCPI standard. It extends *Number* by adding three special values:
	DEF sets the parameter to its default value.
	MIN sets the parameter to its minimum value.
	MAX sets the parameter to its maximum value.
		`-123`
		`12.34`
		`1.234E+01`
		`DEF`
		`MIN`
		`MAX`

Boolean	This is used for system settings that have two states: on / off (or true / false).
	The numeric values 0 and 1 are permitted to indicate off / false and on / true respectively. In addition to numeric values, it is possible to program Boolean settings using the strings **OFF** and **ON**.
	Note that queries return 0 or 1, never 'ON' or 'OFF'.
		`0(false)`
		`1 (true)`
		`OFF (false)`
		`ON (true)`
		`FALSE (error - only 'OFF' and 'ON' are allowed)`

String	String Program Data
	Strings are enclosed in either single (ASCII value 39) or double quotation characters (ASCII value 34). To include one of these characters within the string it must be doubled.
		`"plain string"`
		`'plain string'`
		`"embedded 'c' character"`
		`"embedded""c""character"`
		`'embedded "c" character'`
		`'embedded''c''character'`

5.3.2 Response Data

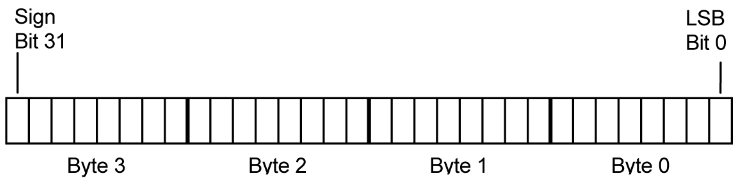
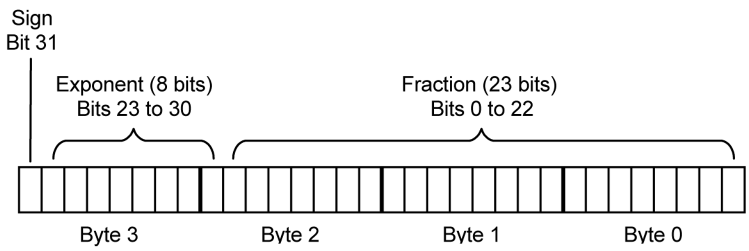
Format	Description	Examples
Selection	This corresponds to the IEEE-488.2 Character Response Data definition and is used for settings that are best described by a string of characters.	AUTO MANUAL
Boolean	Boolean Response Data	0 1 ON and OFF can be used to program a setting but query responses are always 1 or 0.
Integer	This corresponds to the IEEE-488.2 <NR1> Numeric Response Data format and is used for integer values.	12 -34
Real number	The Real number format covers the IEEE-488.2 <NR2> and <NR3> numeric response data formats (explicit radix point numeric values and scaled explicit radix point numeric values with exponent notation).	12.34 -5.678 1.23E+2 -0.123E-6
Binary	This instrument uses only one of the binary data formats defined in IEEE-488.2 for returning graphical measurement data: namely Definite Length Arbitrary Block Response Data. The first character of the response is always '#'. This is followed by a single ASCII digit representing the number of characters needed to represent the length of the binary data. That number of ASCII digits follows. These digits are interpreted as a decimal numeric value corresponding to the length of the binary data. The most significant digit appears first. The data bytes then follow.	#12<0x12><0x31> # marks the start. The length field has 1 digit. The value of the length field is 2 The length field is followed by two bytes (shown here as having the values 12 and 31). #212<0x00>…<0x0B> In this example the length field has two digits, 1 and 2. These are followed by 12 bytes of binary data.
String	String Response Data is identical to string program data. Strings are enclosed in either single (ASCII value 39) or double quotation characters (ASCII value 34). To include one of these characters within the string it must be doubled.	"plain string" 'plain string' "embedded 'c' character" "embedded ""c"" character" 'embedded "c" character' 'embedded ''c'' character'

5 Remote Operation

5.3.3 Binary Header Information

#	Indicates the start of binary data.
n	ASCII digit (1 to 9) giving the number of characters needed to specify the length of the binary data.
mm…m	ASCII digits specifying the length of the binary data in bytes.
	Example: #3256<binary data> '#' Marks the start. '3' is the number of characters needed to represent the length of the data. '256' is the length of the binary data that follows in bytes. Note that the header must always be decoded to determine the length of the data. (The length may change in a future software upgrade.)

5.3.4 Binary Data Fields

INT32	Four bytes interpreted as 32 bit signed integer. Least significant byte first. 
REAL32	Four bytes interpreted as an IEEE 754 standard single precision floating point number. 

5.4 Command Details

This section provides full details of each command. The commands are listed in alphabetical order.

The following information is provided for each command (when applicable).

ABORt:SRWireless	Command name
Function	What the command does.
Command	The command itself.
Query	The request form of the command.
Response	The nature of the response.
Parameters	Parameter definitions.
Example of use	An example of usage.

:ABORt:SRWireless

Function

Abort capture and analysis.

Command

`:ABORt:SRWireless`

Query

N/A

Response

N/A

Parameters

N/A

Example of Use

`:ABOR:SRW`

:CALC:SRWireless

Function

Perform analysis on existing captured data.

Command

```
:CALC:SRWireless
```

Query

N/A

Response

N/A

Parameters

N/A

Example of Use

```
:CALC:SRW
```

:CALCulate:SRWireless:IQ:SAVE

Function

Save the captured IQ data in a file on the instrument in binary format for later retrieval using FTP.

Filenames are generated automatically, and take the form:
IQdataCapture<n>.bin where <n> is an integer representing the current system time.

Files are placed in the following directory: /user/data/

Notes:

Keep an ordered list of the instrument current settings and signal details when saving the IQ data. Later it will be possible to associate the saved files with items on the list by referring to the incrementing timestamps.

IQ data cannot be saved if there is insufficient disc space. Be sure to use your FTP client to delete files from the instrument after you have transferred them.

Command

```
:CALCulate:SRWireless:IQ:SAVE
```

Query

N/A

Response

N/A

Parameters

N/A

Example of Use

```
:CALC:SRW:IQ:SAVE
```

:CONFigure:SRWireless:ALEVel:TIME

Function

Set the capture time used during each stage of the automatic leveling process.

Note:

The automatic leveling function performs up to three measurement captures, so to optimise measurement speed the capture time should be as short as possible. The optimum time depends on the input signal and must be long enough for the system to make an assessment of peak signal power. The default time is 5 ms, which should work well in most cases.

Command

```
:CONFigure:SRWireless:ALEVel:TIME
```

Query

```
:CONFigure:SRWireless:ALEVel:TIME?
```

Response

#	Description	Format	Units	Resolution
1	Automatic leveling capture time	Real number	s	1 µs

Parameters

#	Description	Format	Units	Min	Max	Def	Resolution
1	Automatic leveling capture time	Number+	s	50 µs	100 ms	5 ms	1 µs

Example of Use

```
:CONF:SRW:ALEV:TIME 0.002          // Set the capture time to 2 ms.
:INIT:SRW:ALEV                     // Start automatic leveling function.
*WAI                               // Wait for completion.
:CONF:SRW:POW?                     // Read back the level that was
                                   // determined by the leveling function

> -12.34
```

5

Remote Operation

:CONFigure:SRWireless:BLE:APOWer:LLIMit

Function

Set or query the BLE average output power lower limit (TRM-LE/CA/BV-01-C).

Command

```
:CONFigure:SRWireless:BLE:APOWer:LLIMit
```

Query

```
:CONFigure:SRWireless:BLE:APOWer:LLIMit?
```

Response

#	Description	Format	Units
1	BLE average output power lower limit	Real number	dBm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Average output power lower limit	Number+	dBm	–100	100	–20	0.01 dB

Example of Use

```
:CONF:SRW:BLE:APOW:LLIM -21
```

:CONFigure:SRWireless:BLE:APOWer:ULIMit

Function

Set or query the BLE average output power upper limit (TRM-LE/CA/BV-01-C)

Command

```
:CONFigure:SRWireless:BLE:APOWer:ULIMit
```

Query

```
:CONFigure:SRWireless:BLE:APOWer:ULIMit?
```

Response

#	Description	Format	Units
1	BLE average output power upper limit	Real number	dBm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Average output power upper limit	Number+	dBm	–100	100	10	0.01 dB

Example of Use

```
:CONF:SRW:BLE:APOW:ULIM 10
```

:CONFigure:SRWireless:BLE:B2LE:F1AVerage:LLIMit

Function

Set or query the 2LE (BLE 2 Mbps) Δf1 average lower limit (TRM-LE/CA/BV-10-C, TRM-LE/CA/BV-11-C).

Command

```
:CONFigure:SRWireless:BLE:B2LE:F1AVerage:LLIMit
```

Query

```
:CONFigure:SRWireless:BLE:B2LE:F1AVerage:LLIMit?
```

Response

#	Description	Format	Units
1	2LE (BLE 2 Mbps) Δf1 average lower limit	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	2LE (BLE 2 Mbps) Δf1 average lower limit	Number+	Hz	0	6.00E+05	4.50E+05	1 Hz

Example of Use

```
:CONF:SRW:BLE:B2LE:F1AV:LLIM 450kHz
```

:CONFigure:SRWireless:BLE:B2LE:F1AVerage:ULIMit

Function

Set or query the 2LE (BLE 2 Mbps) Δf1 average upper limit (TRM-LE/CA/BV-10-C, TRM-LE/CA/BV-11-C).

Command

```
:CONFigure:SRWireless:BLE:B2LE:F1AVerage:ULIMit
```

Query

```
:CONFigure:SRWireless:BLE:B2LE:F1AVerage:ULIMit?
```

Response

#	Description	Format	Units
1	2LE (BLE 2 Mbps) Δf1 average upper limit	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	2LE (BLE 2 Mbps) Δf1 average upper limit	Number+	Hz	0	6.00E+05	5.50E+05	1 Hz

Example of Use

```
:CONF:SRW:BLE:B2LE:F1AV:ULIM 550kHz
```

:CONFigure:SRWireless:BLE:B2LE:F21Ratio:LLIMit

Function

Set or query the 2LE (BLE 2 Mbps) Δf2 average / Δf1 average lower limit (TRM-LE/CA/BV-10-C, TRM-LE/CA/BV-11-C).

Command

```
:CONFigure:SRWireless:BLE:B2LE:F21Ratio:LLIMit
```

Query

```
:CONFigure:SRWireless:BLE:B2LE:F21Ratio:LLIMit?
```

Response

#	Description	Format	Units
1	2LE (BLE 2 Mbps) Δf2 average / Δf1 average lower limit	Real number	-

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	2LE (BLE 2 Mbps) Δf2 average / Δf1 average lower limit	Number+	-	0	2.00	0.80	0.01

Example of Use

```
:CONF:SRW:BLE:B2LE:F21R:LLIM 0.8
```

:CONFigure:SRWireless:BLE:B2LE:F2Maximum:LLIMit

Function

Set or query the 2LE (BLE 2 Mbps) Δf2 maximum lower limit (TRM-LE/CA/BV-10-C, TRM-LE/CA/BV-11-C).

Command

```
:CONFigure:SRWireless:BLE:B2LE:F2Maximum:LLIMit
```

Query

```
:CONFigure:SRWireless:BLE:B2LE:F2Maximum:LLIMit?
```

Response

#	Description	Format	Units
1	2LE (BLE 2 Mbps) Δf2 maximum lower limit	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	2LE (BLE 2 Mbps) Δf2 maximum lower limit	Number+	Hz	0	6.00E+05	3.70E+05	1 Hz

Example of Use

```
:CONF:SRW:BLE:B2LE:F2M:LLIM 370kHz
```

:CONFigure:SRWireless:BLE:B2LE:IBEMissions:ULIMit

Function

Set or read the BLE (2 Mbps) in-band emissions measurement upper limits (TRM-LE/CA/BV-08-C).

Command

```
:CONFigure:SRWireless:BLE:B2LE:IBEMissions:ULIMit
```

Query

```
:CONFigure:SRWireless:BLE:B2LE:IBEMissions:ULIMit?
```

Response

#	Description	Format	Units	Resolution
1	Limit for carrier frequency ± 4 MHz	Real number	dB	0.01 dB
2	Limit for carrier frequency ± 5 MHz	Real number	dB	0.01 dB
3	Limit for carrier frequency ± 6 MHz, 7 MHz and 8 MHz	Real number	dB	0.01 dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Limit for carrier frequency ± 4 MHz	Number	dB	–100.00	20.00	–20.00	0.01 dB
2	Limit for carrier frequency ± 5 MHz	Number	dB	-100.00	20.00	-20.00	0.01 dB
2	Limit for carrier frequency ± 6 MHz, 7 MHz and 8 MHz	Number	dB	–100.00	20.00	–30.00	0.01 dB

Example of Use

```
:CONF:SRW:BLE:B2LE:IBEM:ULIM -22,-22,-35
                              // Set the emissions limit to -22 dB
                              // at ±4 MHz from the carrier,
                              // -22 dB at ±5 MHz from the carrier
                              // and -35 dB at ±6 MHz, 7 MHz and
                              // 8 MHz from the carrier.
```

:CONFigure:SRWireless:BLE:BLR:CDRift:LIMit

Function

Set or query the BLR (BLE coded S=8) carrier frequency drift limit (±) (TRM-LE/CA/BV-14-C).

Command

```
:CONFigure:SRWireless:BLE:BLR:CDRift:LIMit
```

Query

```
:CONFigure:SRWireless:BLE:BLR:CDRift:LIMit?
```

Response

#	Description	Format	Units
1	BLR (BLE coded S=8) carrier frequency drift limit (±)	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLR (BLE coded S=8) carrier frequency drift limit (±)	Number +	Hz	0	2.00E+05	50000	1 Hz

Example of Use

```
:CONF:SRW:BLE:BLR:CDR:LIM 45000     // Set the carrier drift limit to ±45 kHz.
```

:CONFigure:SRWireless:BLE:BLR:CFOFfset:LIMit

Function

Set or query the BLR (BLE coded S=8) carrier frequency offset limit (±) (TRM-LE/CA/BV-14-C).

Command

```
:CONFigure:SRWireless:BLE:BLR:CFOFfset:LIMit
```

Query

```
:CONFigure:SRWireless:BLE:BLR:CFOFfset:LIMit?
```

Response

#	Description	Format	Units
1	BLR (BLE coded S=8) carrier frequency offset limit (±)	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLR (BLE coded S=8) carrier frequency offset limit (±)	Number+	Hz	0	2.00E+05	150000	1 Hz

Example of Use

```
:CONF:SRW:BLE:BLR:CFOF:LIM 145000   // Set the limit to ±145 kHz
```

:CONFigure:SRWireless:BLE:BLR:DRATe:LIMit

Function

Set or query the BLR (BLE coded S=8) carrier frequency drift rate limit (±) (TRM-LE/CA/BV-14-C).

Command

```
:CONFigure:SRWireless:BLE:BLR:DRATe:LIMit
```

Query

```
:CONFigure:SRWireless:BLE:BLR:DRATe:LIMit?
```

Response

#	Description	Format	Units
1	BLR (BLE coded S=8) carrier frequency drift rate limit (±)	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLR (BLE coded S=8) carrier frequency drift rate limit (±)	Number+	Hz	0	2.00E+05	19200	1 Hz

Example of Use

```
:CONF:SRW:BLE:BLR:DRAT:LIM 19000    // Set the drift rate limit to ±19 kHz
```

:CONFigure:SRWireless:BLE:BLR:F1AVerage:LLIMit

Function

Set or query the BLR (BLE coded S=8) Δf1 average lower limit (TRM-LE/CA/BV-13-C).

Command

```
:CONFigure:SRWireless:BLE:BLR:F1AVerage:LLIMit
```

Query

```
:CONFigure:SRWireless:BLE:BLR:F1AVerage:LLIMit?
```

Response

#	Description	Format	Units
1	BLR (BLE coded S=8) Δf1 average lower limit	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLR (BLE coded S=8) Δf1 average lower limit	Number+	Hz	0	3.00E+05	2.25E+05	1 Hz

Example of Use

```
:CONF:SRW:BLE:BLR:F1AV:LLIM 225kHz
```

:CONFigure:SRWireless:BLE:BLR:F1AVerage:ULIMit

Function

Set or query the BLR (BLE coded S=8) Δf1 average upper limit (TRM-LE/CA/BV-13-C).

Command

```
:CONFigure:SRWireless:BLE:BLR:F1AVerage:ULIMit
```

Query

```
:CONFigure:SRWireless:BLE:BLR:F1AVerage:ULIMit?
```

Response

#	Description	Format	Units
1	BLR (BLE coded S=8) Δf1 average upper limit	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLR (BLE coded S=8) Δf1 average upper limit	Number+	Hz	0	3.00E+05	2.75E+05	1 Hz

Example of Use

```
:CONF:SRW:BLE:BLR:F1AV:ULIM 275kHz
```

5

Remote Operation

:CONFigure:SRWireless:BLE:BLR:F1Maximum:LLIMit

Function

Set or query the BLR (BLE coded S=8) Δf1 maximum lower limit (TRM-LE/CA/BV-13-C).

Command

```
:CONFigure:SRWireless:BLE:BLR:F1Maximum:LLIMit
```

Query

```
:CONFigure:SRWireless:BLE:BLR:F1Maximum:LLIMit?
```

Response

#	Description	Format	Units
1	BLR (BLE coded S=8) Δf1 maximum lower limit	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLR (BLE coded S=8) Δf1 maximum lower limit	Number+	Hz	0	3.00E+05	1.85E+05	1 Hz

Example of Use

```
:CONF:SRW:BLE:BLR:F1M:LLIM 185kHz
```

:CONFigure:SRWireless:BLE:BLR:IDRate:LIMit

Function

Set or query the BLR (BLE coded S=8) initial carrier frequency drift rate limit (±) (TRM-LE/CA/BV-14-C).

Command

```
:CONFigure:SRWireless:BLE:BLR:IDRate:LIMit
```

Query

```
:CONFigure:SRWireless:BLE:BLR:IDRate:LIMit?
```

Response

#	Description	Format	Units
1	BLR (BLE coded S=8) initial carrier frequency drift rate limit (±)	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLR (BLE coded S=8) initial carrier frequency drift limit (±)	Number+	Hz	0	2.00E+05	19200	1 Hz

Example of Use

```
:CONF:SRW:BLE:BLR:IDR:LIM 19000     // Set the initial carrier drift rate
                                    // limit to ±19 kHz.
```

5 Remote Operation

:CONFigure:SRWireless:BLE:CDRift:LIMit

Function

Set or query the BLE(1 Mbps and 2 Mbps) carrier frequency drift limit (±) (TRM-LE/CA/BV-06-C and TRM-LE/CA/BV-12-C).

Command

```
:CONFigure:SRWireless:BLE:CDRift:LIMit
```

Query

```
:CONFigure:SRWireless:BLE:CDRift:LIMit?
```

Response

#	Description	Format	Units
1	BLE carrier frequency drift limit (±)	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLE carrier frequency drift limit (±)	Number+	Hz	0	2.00E+05	50000	1 Hz

Example of Use

```
:CONF:SRW:BLE:CDR:LIM 45000        // Set the carrier drift limit to ±45 kHz.
```

:CONFigure:SRWireless:BLE:CFACtor:ULIMit

Function

Set or query the BLE crest factor upper limit (TRM-LE/CA/BV-01-C).

Command

```
:CONFigure:SRWireless:BLE:CFACtor:ULIMit
```

Query

```
:CONFigure:SRWireless:BLE:CFACtor:ULIMit?
```

Response

#	Description	Format	Units
1	BLE crest factor upper limit	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLE crest factor upper limit	Number+	dB	–100	100	3	0.01

Example of Use

```
:CONF:SRW:BLE:CFAC:ULIM 3.5
```

:CONFigure:SRWireless:BLE:CFOFfset:LIMit

Function

Set or query the BLE (1 Mbps and 2 Mbps) carrier frequency offset limit (±) (TRM-LE/CA/BV-06-C and TRM-LE/CA/BV-12-C).

Command

```
:CONFigure:SRWireless:BLE:CFOFfset:LIMit
```

Query

```
:CONFigure:SRWireless:BLE:CFOFfset:LIMit?
```

Response

#	Description	Format	Units
1	BLE carrier frequency offset limit (±)	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLE carrier frequency offset limit (±)	Number+	Hz	0	2.00E+05	150000	1 Hz

Example of Use

```
:CONF:SRW:BLE:CFOF:LIM 155000       // Set the limit to ±155 kHz
```

:CONFigure:SRWireless:BLE:DRATe:LIMit

Function

Set or query the BLE (1 Mbps and 2 Mbps) carrier frequency drift rate limit (±) (TRM-LE/CA/BV-06-C and TRM-LE/CA/BV-12-C).

Command

```
:CONFigure:SRWireless:BLE:DRATe:LIMit
```

Query

```
:CONFigure:SRWireless:BLE:DRATe:LIMit?
```

Response

#	Description	Format	Units
1	BLE carrier frequency drift rate limit (±)	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLE carrier frequency drift rate limit (±)	Number+	Hz	0	2.00E+05	20000	1 Hz

Example of Use

```
:CONF:SRW:BLE:DRAT:LIM 19000        // Set the drift rate to ±19 kHz
```

:CONFigure:SRWireless:BLE:F1AVerage:LLIMit

Function

Set or query the BLE Δf1 (1 Mbps) average lower limit (TRM-LE/CA/BV-05-C, TRM-LE/CA/BV-09-C).

Command

```
:CONFigure:SRWireless:BLE:F1AVerage:LLIMit
```

Query

```
:CONFigure:SRWireless:BLE:F1AVerage:LLIMit?
```

Response

#	Description	Format	Units
1	BLE Δf1 average lower limit	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLE Δf1 average lower limit	Number+	Hz	0	3.00E+05	2.25E+05	100 Hz

Example of Use

```
:CONF:SRW:BLE:F1AV:LLIM 230kHz
```

:CONFigure:SRWireless:BLE:F1AVerage:ULIMit

Function

Set or query the BLE (1 Mbps) Δf1 average upper limit (TRM-LE/CA/BV-05-C, TRM-LE/CA/BV-09-C).

Command

```
:CONFigure:SRWireless:BLE:F1AVerage:ULIMit
```

Query

```
:CONFigure:SRWireless:BLE:F1AVerage:ULIMit?
```

Response

#	Description	Format	Units
1	BLE Δf1 average upper limit	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLE Δf1 average upper limit	Number+	Hz	0	3.00E+05	2.75E+05	100 Hz

Example of Use

```
:CONF:SRW:BLE:F1AV:ULIM 280kHz
```

:CONFigure:SRWireless:BLE:F21Ratio:LLIMit

Function

Set or query the BLE (1 Mbps) Δf2 average / f1 average lower limit (TRM-LE/CA/BV-05-C, TRM-LE/CA/BV-09-C).

Command

```
:CONFigure:SRWireless:BLE:F21Ratio:LLIMit
```

Query

```
:CONFigure:SRWireless:BLE:F21Ratio:LLIMit?
```

Response

#	Description	Format	Units
1	BLE Δf2 average / Δf1 average lower limit	Real number	-

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLE Δf2 average / Δf1 average lower limit	Number+	-	0	2.00	0.8E–01	0.01

Example of Use

```
:CONF:SRW:BLE:F21R:LLIM 0.8
```

:CONFigure:SRWireless:BLE:F2Maximum:LLIMit

Function

Set or query the BLE (1 Mbps) Δf2 maximum lower limit (TRM-LE/CA/BV-05-C, TRM-LE/CA/BV-09-C).

Command

```
:CONFigure:SRWireless:BLE:F2Maximum:LLIMit
```

Query

```
:CONFigure:SRWireless:BLE:F2Maximum:LLIMit?
```

Response

#	Description	Format	Units
1	BLE Δf2 maximum lower limit	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLE Δf2 maximum lower limit	Number+	Hz	0	2.50E+05	1.85E+05	100 Hz

Example of Use

```
:CONF:SRW:BLE:F2M:LLIM 190E+03
```

:CONFigure:SRWireless:BLE:IBEMissions:ULIMit

Function

Set or read the BLE (1 Mbps) in-band emissions measurement upper limits (TRM-LE/CA/BV-03-C).

Command

```
:CONFigure:SRWireless:BLE:IBEMissions:ULIMit
```

Query

```
:CONFigure:SRWireless:BLE:IBEMissions:ULIMit?
```

Response

#	Description	Format	Units	Resolution
1	Limit for carrier frequency ± 2 MHz	Real number	dB	0.01 dB
2	Limit for carrier frequency ± 3 MHz, 4 MHz and 5 MHz	Real number	dB	0.01 dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Limit for carrier frequency ± 2 MHz	Number	dB	–100.00	20.00	–20.00	0.01 dB
2	Limit for carrier frequency ± 3 MHz, 4 MHz and 5 MHz	Number	dB	–100.00	20.00	–30.00	0.01 dB

Example of Use

```
:CONF:SRW:BLE:IBEM:ULIM -15,-20     // Set the emissions limit to -15 dB
                                    // at ± 2 MHz from the carrier
                                    // and -20 dB at ± 3 MHz, 4 MHz and
                                    // 5 MHz from the carrier.
```

:CONFigure:SRWireless:BLE:IDRate:LIMit

Function

Set or query the BLE (1 Mbps and 2 Mbps) initial carrier frequency drift rate limit (±) (TRM-LE/CA/BV-06-C and TRM-LE/CA/BV-12-C).

Command

```
:CONFigure:SRWireless:BLE:IDRate:LIMit
```

Query

```
:CONFigure:SRWireless:BLE:IDRate:LIMit?
```

Response

#	Description	Format	Units
1	BLE initial carrier frequency drift rate limit (±)	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BLE initial carrier frequency drift limit (±)	Number+	Hz	0	2.00E+05	23000	1 Hz

Example of Use

```
:CONF:SRW:BLE:IDR:LIM 20000        // Set the carrier drift limit to ±20 kHz.
```

:CONFigure:SRWireless:BLE:LRCoding

Function

Set or query the expected coding scheme for the *Bluetooth* Low Energy Long Range segment.

Note:
This setting is ignored if the expected packet type for the segment is not BLR.
See also :CONFigure:SRWireless:BLE:PACKet

Command

```
:CONFigure:SRWireless:BLE:LRCoding
```

Query

```
:CONFigure:SRWireless:BLE:LRCoding?
```

Response

#	Description	Format
1	BLR coding scheme	<CRD>

Parameters

#	Description	Format	Units	Min	Max	Default
1	BLR coding scheme	See below	---	---	---	S8

Bluetooth Low Energy packet type:

S8	Low energy coded (S=8)
S2	Low energy coded (S=2)

Example of Use

```
:CONF:SRW:BLE:LRC S2 // Set the expected coding scheme to S=2.
```

:CONFigure:SRWireless:BLE:MODE (Deprecated)

Function

Set or query the *Bluetooth* Low Energy measurement mode: “SIG Standard” or “Speed Test”.

Note:

This command is deprecated since the addition of the AUTOBT wireless standard, and should not be used for new designs. Instead use `:CONFigure:SRWireless:BT:MODE` when configuring BT, BLE or AUTOBT segments.

Command

```
:CONFigure:SRWireless:BLE:MODE
```

Query

```
:CONFigure:SRWireless:BLE:MODE?
```

Response

#	Description	Format
1	*Bluetooth* (BR & EDR) measurement mode	<CRD>

Parameters

#	Description	Format	Units	Min	Max	Default
1	*Bluetooth* Low Energy measurement mode	See below	---	---	---	SPEED

Bluetooth (BR & EDR) measurement mode:

STANDARD	*Bluetooth* basic rate and EDR measurements are made according to the SIG Standard. In this mode certain measurements are only made when the payload type is as specified by the *Bluetooth* standard. For this reason it is not possible to extract all measurements from a single capture segment.
SPEED	*Bluetooth* basic rate and EDR measurements are made in Speed Test mode. In this mode all measurements can be made regardless of the payload type.

Example of Use

```
:CONF:SRW:BLE:MODE STANDARD
```

5 Remote Operation

:CONFigure:SRWireless:BLE:PACKet

Function

Set or query the expected packet type of the *Bluetooth* Low Energy packets in this segment.

Command

```
:CONFigure:SRWireless:BLE:PACKet
```

Query

```
:CONFigure:SRWireless:BLE:PACKet?
```

Response

#	Description	Format
1	BLE packet type	<CRD>

Parameters

#	Description	Format	Units	Min	Max	Default
1	BLE packet type	See below	---	---	---	BLE

Bluetooth Low Energy packet type:

BLE	Low energy 1 Mbps
2LE	Low energy 2 Mbps
BLR	Low energy long range (coded)

Note:
Use :CONfigure:SRWireless:BLR:LRCoding to set the coding scheme for long range packets: S=8 or S=2

Example of Use

```
:CONF:SRW:BLE:PACK BLR       // Set the expected packet type to long range
```

:CONFigure:SRWireless:BLE:PAYLoad

Function

Set or query the expected payload type of the *Bluetooth* Low Energy packets in this segment.

Command

```
:CONFigure:SRWireless:BLE:PAYLoad
```

Query

```
:CONFigure:SRWireless:BLE:PAYLoad?
```

Response

#	Description	Format
1	BLE payload	<CRD>

Parameters

#	Description	Format	Units	Min	Max	Default
1	BLE payload	See below	---	---	---	PRBS9

Bluetooth Low Energy payload type:

AAHEX	Payload is 10101010 (0xAA)
FZEROHEX	Payload is 11110000 (0xF0)
PRBS9	Payload is a pseudorandom binary sequence .
ONES	Payload is 11111111 (0xFF) Used for BLR (S=8) modulation characteristics and carrier offset and drift measurements. After passing through the S=8 encoder, this sequence become a repetitive sequence of 00111100 symbols.

Example of Use

```
:CONF:SRW:BLE:PAYL FZEROHEX        // Set the expected payload to 11110000
```

:CONFigure:SRWireless:BLE:PLENgth

Function

Sets the expected *Bluetooth* low energy packet length.

Command

```
:CONFigure:SRWireless:BLE:PLENgth
```

Query

```
:CONFigure:SRWireless:BLE:PLENgth?
```

Response

#	Description	Format
1	Bluetooth low energy packet length in bytes	Integer

Parameters

#	Description	Format	Min	Max	Default
1	Bluetooth low energy packet length in bytes	Number+	0	255*1	37

*1: The value is limited to 37 bytes if there is no MX887040A-001 installed.

Example of Use

```
:CONF:SRWireless:BLE:PLENgth 255    // Set the payload length
                                    // to 255 bytes.
```

:CONFigure:SRWireless:BT:ACPower:ULIMit

Function

Set the *Bluetooth* adjacent channel power upper limits (TRM/CA/06/C).

Command

```
:CONFigure:SRWireless:BT:ACPower:ULIMit
```

Query

```
:CONFigure:SRWireless:BT:ACPower:ULIMit?
```

Response

#	Description	Format	Units
1	ACP upper limit for channel ±2	Real number	dBm
2	ACP upper limit for channels ±3, 4, 5	Real number	dBm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	ACP upper limit for channel ±2	Number+	dBm	–100	+20	–20	0.01 dB
2	ACP upper limit for channels ±3, 4, 5	Number+	dBm	–100	+20	–40	0.01 dB

Example of Use

```
:CONF:SRW:BT:ACP:ULIM -25,-45
```

:CONFigure:SRWireless:BT:ADDRess

Function

Set or query the *Bluetooth* address for basic rate, EDR, and low energy.

Command

```
:CONFigure:SRWireless:BT:ADDRess
```

Query

```
:CONFigure:SRWireless:BT:ADDRess?
```

Response

#	Description	Format
1	*Bluetooth* address	String

Parameters

#	Description	Format	Units	Min	Max	Default
1	*Bluetooth* address	String	---	---	---	"000000C6967E"

Example of Use

```
:CONF:SRW:BT:ADDR?
➢ "000000C6967E"
```

:CONFigure:SRWireless:BT:APOWer:LLIMit

Function

Set or query the *Bluetooth* average output power lower limit (TRM/CA/01/C).

Command

```
:CONFigure:SRWireless:BT:APOWer:LLIMit
```

Query

```
:CONFigure:SRWireless:BT:APOWer:LLIMit?
```

Response

#	Description	Format	Units
1	Average output power lower limit	Real number	dBm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Average output power lower limit	Number+	dBm	–100	100	–6	0.01 dB

Example of Use

```
:CONF:SRW:BT:APOW:LLIM -5           // Set limit to -5 dBm
```

:CONFigure:SRWireless:BT:APOWer:ULIMit

Function

Set or query the *Bluetooth* average output power upper limit (TRM/CA/01/C)

Command

```
:CONFigure:SRWireless:BT:APOWer:ULIMit
```

Query

```
:CONFigure:SRWireless:BT:APOWer:ULIMit?
```

Response

#	Description	Format	Units
1	Average output power upper limit	Real number	dBm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Average output power upper limit	Number+	dBm	–100	100	20	0.01 dB

Example of Use

```
:CONF:SRW:BT:APOW:ULIM 19          // Set limit to 19 dBm
```

:CONFigure:SRWireless:BT:BANDwidth:ULIMit

Function

Set the *Bluetooth* 20 dB bandwidth measurement upper limit (TRM/CA/05/C).

Command

```
:CONFigure:SRWireless:BT:BANDwidth:ULIMit
```

Query

```
:CONFigure:SRWireless:BT:BANDwidth:ULIMit?
```

Response

#	Description	Format	Units
1	Bluetooth 20 dB bandwidth measurement upper limit	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Bluetooth 20 dB bandwidth measurement upper limit	Number+	Hz	800 kHz	2 MHz	1 MHz	1 Hz

Example of Use

```
:CONF:SRW:BT:BAND:ULIM 1500000      // Set the bandwidth limit to 1.5 MHz.
```

:CONFigure:SRWireless:BT:CDRift:LIMit

Function

Set or query the *Bluetooth* carrier frequency drift limit (±) (TRM/CA/09/C)

Command

```
:CONFigure:SRWireless:BT:CDRift:LIMit
```

Query

```
:CONFigure:SRWireless:BT:CDRift:LIMit?
```

Response

#	Description	Format	Units
1	Drift limit (±) – DH1 packet types	Real number	Hz
2	Drift limit (±) – DH3 packet types	Real number	Hz
3	Drift limit (±) – DH5 packet types	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Drift limit (±) – DH1 packet types	Number+	Hz	0	200000	25000	1 Hz
2	Drift limit (±) – DH3 packet types	Number+	Hz	0	200000	40000	1 Hz
3	Drift limit (±) – DH5 packet types	Number+	Hz	0	200000	40000	1 Hz

Example of Use

```
:CONF:SRW:BT:CDR:LIM 20000,35000,38000     // Set drift limits:
                                           // ±20 kHz for DH1 packets
                                           // ±35 kHz for DH3 packets
                                           // ±38 kHz for DH5 packets
```

:CONFigure:SRWireless:BT:CFSTability:LIMit

Function

Bluetooth EDR carrier frequency stability limits (±) (TRM/CA/11/C)

Command

```
:CONFigure:SRWireless:BT:CFSTability:LIMit
```

Query

```
:CONFigure:SRWireless:BT:CFSTability:LIMit?
```

Response

#	Description	Format	Units
1	Block frequency error limit ($\omega_i + \omega_0$) (±)	Real number	Hz
2	Initial frequency error limit (ω_i) (±)	Real number	Hz
3	Frequency error limit (ω_0) (±)	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Block frequency error limit ($\omega_i + \omega_0$) (±)	Number+	Hz	0	200000	75000	1 Hz
2	Initial frequency error limit (ω_i) (±) (Optional)	Number+	Hz	0	200000	75000	1 Hz
3	Frequency error limit (ω_0) (±) (Optional)	Number+	Hz	0	200000	10000	1 Hz

Note:

The second and third parameters are optional to ensure backward compatibility with earlier versions of the Short Range Wireless Application.

Example of Use

```
:CONF:SRW:BT:CFST:LIM 70000,70000,9000
```

:CONFigure:SRWireless:BT:DRATe:LIMit

Function

Set or query the *Bluetooth* carrier frequency drift rate limit (±) (TRM/CA/09/C).

Command

```
:CONFigure:SRWireless:BT:DRATe:LIMit
```

Query

```
:CONFigure:SRWireless:BT:DRATe:LIMit?
```

Response

#	Description	Format	Units
1	Carrier frequency drift rate limit (±)	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Carrier frequency drift rate limit (±)	Number+	Hz	0	200000	20000	1 Hz

Example of Use

```
:CONF:SRW:BT:DRAT:LIM 18kHz
```

:CONFigure:SRWireless:BT:EPER:ULIMit

Function

Set the *Bluetooth* EDR differential phase encoding packet error rate (PER) upper limit (TRM/CA/12/C).

Command

```
:CONFigure:SRWireless:BT:EPER:ULIMit
```

Query

```
:CONFigure:SRWireless:BT:EPER:ULIMit?
```

Response

#	Description	Format	Units
1	BT EDR differential phase encoding PER upper limit	Real number	%

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	BT EDR differential phase encoding PER upper limit	Number+	%	0	100	1.00	0.01%

Example of Use

```
:CONF:SRW:BT:EPER:ULIM 2            // Set the limit to 2%
```

:CONFigure:SRWireless:BT:ERTPower:LLIMit

Function

Set or query the *Bluetooth* EDR relative transmit power lower limit (TRM/CA/10/C).

Command

```
:CONFigure:SRWireless:BT:ERTPower:LLIMit
```

Query

```
:CONFigure:SRWireless:BT:ERTPower:LLIMit?
```

Response

#	Description	Format	Units
1	EDR relative transmit power lower limit	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	EDR relative transmit power lower limit	Number+	dB	–100	100	–4	0.01 dB

Example of Use

```
:CONF:SRW:BT:ERTP:LLIM -1.00
```

:CONFigure:SRWireless:BT:ERTPower:ULIMit

Function

Set or query the *Bluetooth* EDR relative transmit power upper limit (TRM/CA/10/C).

Command

```
:CONFigure:SRWireless:BT:ERTPower:ULIMit
```

Query

```
:CONFigure:SRWireless:BT:ERTPower:ULIMit?
```

Response

#	Description	Format	Units
1	EDR relative transmit power upper limit	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	EDR relative transmit power upper limit	Number+	dB	–100	100	1	0.01 dB

Example of Use

```
:CONF:SRW:BT:ERTP:ULIM 0.8
```

:CONFigure:SRWireless:BT:F1AVerage:LLIMit

Function

Set or query the *Bluetooth* Δf1 average lower limit (TRM/CA/07/C).

Command

```
:CONFigure:SRWireless:BT:F1AVerage:LLIMit
```

Query

```
:CONFigure:SRWireless:BT:F1AVerage:LLIMit?
```

Response

#	Description	Format	Units
1	Δf1 average lower limit	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Δf1 average lower limit	Number+	Hz	0	200000	140000	1 Hz

Example of Use

```
:CONF:SRW:BT:F1AV:LLIM 135kHz
```

:CONFigure:SRWireless:BT:F1AVerage:ULIMit

Function

Set or query the *Bluetooth* delta f1 average upper limit (TRM/CA/07/C)

Command

```
:CONFigure:SRWireless:BT:F1AVerage:ULIMit
```

Query

```
:CONFigure:SRWireless:BT:F1AVerage:ULIMit?
```

Response

#	Description	Format	Units
1	Δf1 average upper limit	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Δf1 average upper limit	Number+	Hz	0	200000	175000	1 Hz

Example of Use

```
:CONF:SRW:BT:F1AV:ULIM 170kHz
```

:CONFigure:SRWireless:BT:F21Ratio:LLIMit

Function

Set or query the *Bluetooth* Δf2 average / Δf1 average lower limit (TRM/CA/07/C).

Command

```
:CONFigure:SRWireless:BT:F21Ratio:LLIMit
```

Query

```
:CONFigure:SRWireless:BT:F21Ratio:LLIMit?
```

Response

#	Description	Format	Units
1	Δf2 average / Δf1 average lower limit	Real number	---

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Δf2 average / Δf1 average lower limit	Number+	---	0	2	0.8	0.01

Example of Use

```
:CONF:SRW:BT:F21R:LLIM 0.75
```

:CONFigure:SRWireless:BT:F2Maximum:LLIMit

Function

Set or query the *Bluetooth* Δf2 maximum lower limit (TRM/CA/07/C)

Command

```
:CONFigure:SRWireless:BT:F2Maximum:LLIMit
```

Query

```
:CONFigure:SRWireless:BT:F2Maximum:LLIMit?
```

Response

#	Description	Format	Units
1	Δf2 maximum lower limit	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Δf2 maximum lower limit	Number+	Hz	0	200000	115000	1 Hz

Example of Use

```
:CONF:SRW:BT:F2M:LLIM 110kHz
```

:CONFigure:SRWireless:BT:FRANge:LIMit

Function

Set/query the *Bluetooth* frequency range measurement limit (TRM/CA/04/C).

Command

```
:CONFigure:SRWireless:BT:FRANge:LIMit
```

Query

```
:CONFigure:SRWireless:BT:FRANge:LIMit?
```

Response

#	Description	Format	Units
1	Lower frequency band limit	Integer	Hz
2	Upper frequency band limit	Integer	Hz

Parameters

#	Description	Format	Units	Min	Max	Default
1	Lower frequency band limit	Number+	Hz	2300000000	2600000000	2400000000
2	Upper frequency band limit	Number+	Hz	2300000000	2600000000	2483500000

Example of Use

```
:CONF:SRW:BT:FRAN:LIM 2.3E+9,2.5E+9
```

:CONFigure:SRWireless:BT:GUARd:LIMit

Function

Set or query the *Bluetooth* EDR guard interval limit (±)

Note:

The limit should be thought of as a guard interval tolerance that is applied to the nominal value of 5 μs. When the limit is set to the default of 0.25 μs, it means that the measured guard interval may be in the range 4.75 μs to 5.25 μs (5 - 0.25 = 4.75 μs and 5 + 0.25 = 5.25 μs).

Command

```
:CONFigure:SRWireless:BT:GUARd:LIMit
```

Query

```
:CONFigure:SRWireless:BT:GUARd:LIMit?
```

Response

#	Description	Format	Units
1	EDR guard interval limit (±)	Real number	s

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	EDR guard interval limit (±)	Number+	s	0	5.0 μs	0.25 μs	0.01 μs

Example of Use

```
:CONF:SRW:BT:GUAR:LIM 2E-06        // Set the limit to ±2 μs.
```

:CONFigure:SRWireless:BT:IBSPurious:ULIMit

Function

Set the *Bluetooth* EDR in band spurious emissions upper limits (TRM/CA/13/C).

Command

```
:CONFigure:SRWireless:BT:IBSPurious:ULIMit
```

Query

```
: CONFigure:SRWireless:BT:IBSPurious:ULIMit?
```

Response

#	Description	Format	Units
1	In band spurious upper limit for channel ±1 (relative to carrier).	Real number	dBc
2	In band spurious upper limit for channel ±2 (absolute power)	Real number	dBm
3	In band spurious upper limit for channels ±3, 4, 5 (absolute power)	Real number	dBm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	In band spurious upper limit for channel ±1 (relative to carrier).	Number+	dBc	–100	+20	–26	0.01 dB
2	In band spurious upper limit for channel ±2 (absolute power)	Number+	dBm	–100	+20	–20	0.01 dB
3	In band spurious upper limit for channels ±3, 4, 5 (absolute power)	Number+	dBm	–100	+20	–40	0.01 dB

Example of Use

```
:CONF:SRW:BT:IBSP:ULIM -27, -25,-45
```

:CONFigure:SRWireless:BT:ICFTolerance:LIMit

Function

Set or query the *Bluetooth* initial carrier frequency tolerance limit (±) (TRM/CA/08/C)

Command

```
:CONFigure:SRWireless:BT:ICFTolerance:LIMit
```

Query

```
:CONFigure:SRWireless:BT:ICFTolerance:LIMit?
```

Response

#	Description	Format	Units
1	Initial carrier frequency tolerance limit (±)	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Initial carrier frequency tolerance limit (±)	Number+	Hz	0	200000	75000	1 Hz

Example of Use

```
:CONF:SRW:BT:ICFT:LIM 70kHz        // Set limit to ±70 kHz
```

:CONFigure:SRWireless:BT:MODE

Function

Set or query the *Bluetooth* measurement mode: “SIG Standard”, “Speed Test” or “Speed Test with Header Exclusion”.

Command

```
:CONFigure:SRWireless:BT:MODE
```

Query

```
:CONFigure:SRWireless:BT:MODE?
```

Response

#	Description	Format
1	*Bluetooth* (BR & EDR) measurement mode	<CRD>

Parameters

#	Description	Format	Units	Min	Max	Default
1	*Bluetooth* (BR & EDR) measurement mode	See below	---	---	---	SPEED

Bluetooth (BR & EDR) measurement mode:

STANDARD	*Bluetooth* basic rate and EDR measurements are made according to the SIG Standard. In this mode certain measurements are only made when the payload type is as specified by the *Bluetooth* standard. For this reason it is not possible to extract all measurements from a single capture segment.
SPEED	*Bluetooth* basic rate and EDR measurements are made in Speed Test mode. In this mode all measurements can be made regardless of the payload type.
HDREXCLUDE	Header exclusion mode is an extension to speed test mode that is available when the wireless standard selected for the segment is AUTOBT. In this mode the system does not rely on the packet header information being correct. Header exclusion mode is not available when the wireless standard is BT or BLE. If it is selected for these segments, or if the segment type is AUTOBT and a BLE packet is received, the system behaves as though speed test alone were selected.

Example of Use

```
:CONF:SRW:BT:MODE STANDARD
```

:CONFigure:SRWireless:BT:PACKet

Function

Set or query the expected packet type of the *Bluetooth* (Basic Rate or EDR) packets in this segment.

Command

```
:CONFigure:SRWireless:BT:PACKet
```

Query

```
:CONFigure:SRWireless:BT:PACKet?
```

Response

#	Description	Format	Units
1	*Bluetooth* (BR & EDR) packet type.	See below	

Parameters

#	Description	Format	Units	Min	Max	Default
1	*Bluetooth* (BR & EDR) packet type.	See below	---	---	---	DH1

Bluetooth (BR & EDR) packet type:

DH1	DH1 (Basic Rate)
DH3	DH3 (Basic Rate)
DH5	DH5 (Basic Rate)
2DH1	2DH1 (2 Mbps EDR)
2DH3	2DH3 (2 Mbps EDR)
2DH5	2DH5 (2 Mbps EDR)
3DH1	3DH1 (3 Mbps EDR)
3DH3	3DH3 (3 Mbps EDR)
3DH5	3DH5 (3 Mbps EDR)

Example of Use

```
:CONF:SRW:BT:PACK 3DH5
```

:CONFigure:SRWireless:BT:PAYLoad

Function

Set or query the expected payload type of the Bluetooth (BR / EDR) packets in this segment.

Command

```
:CONFigure:SRWireless:BT:PAYLoad
```

Query

```
:CONFigure:SRWireless:BT:PAYLoad?
```

Response

#	Description	Format	Units
1	*Bluetooth* (BR & EDR) payload type	See below	

Parameters

#	Description	Format	Units	Min	Max	Default
1	*Bluetooth* (BR & EDR) payload type	See below	---	---	---	PRBS9

Bluetooth (BR & EDR) payload type:

AAHEX	Payload is 10101010 (0xAA)
FZEROHEX	Payload is 11110000 (0xF0)
PRBS9	Payload is a pseudorandom binary sequence .

Example of Use

```
:CONF:SRW:BT:PAYL AAHEX      // Set the expected payload to 10101010
```

:CONFigure:SRWireless:BT:PCDevm:ULIMit

Function

Set or query the *Bluetooth* EDR DEVM limit to be achieved by 99% of symbols (TRM/CA/11/C)

Command

```
:CONFigure:SRWireless:BT:PCDevm:ULIMit
```

Query

```
:CONFigure:SRWireless:BT:PCDevm:ULIMit?
```

Response

#	Description	Format	Units
1	EDR peak DEVM upper limit which 99% of packets must pass (π/4-DQPSK blocks)	Real number	-
2	EDR peak DEVM upper limit which 99% of packets must pass (8DPSK blocks)	Real number	-

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	EDR peak DEVM upper limit which 99% of packets must pass (π/4-DQPSK blocks)	Number+	-	0	1	0.3	0.001
2	EDR peak DEVM upper limit which 99% of packets must pass (8DPSK blocks)	Number+	-	0	1	0.2	0.001

Example of Use

```
:CONF:SRW:BT:PCD:ULIM 0.35,0.25
```

:CONFigure:SRWireless:BT:PDEVm:ULIMit

Function

Set or query the *Bluetooth* EDR Peak DEVM upper limit (TRM/CA/11/C)

Command

```
:CONFigure:SRWireless:BT:PDEVm:ULIMit
```

Query

```
:CONFigure:SRWireless:BT:PDEVm:ULIMit?
```

Response

#	Description	Format	Units
1	EDR peak DEVM upper limit (π/4-DQPSK blocks)	Real number	---
2	EDR peak DEVM upper limit (8DPSK blocks)	Real number	---

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	EDR peak DEVM upper limit (π/4-DQPSK blocks)	Number+	---	0	1	0.35	0.001
2	EDR peak DEVM upper limit (8DPSK blocks)	Number+	---	0	1	0.25	0.001

Example of Use

```
:CONF:SRW:BT:PDEV:ULIM 0.35,0.25
```

:CONFigure:SRWireless:BT:PLENgth

Function

Set or query the expected payload length of the *Bluetooth* (BR / EDR) packets in this segment.

Command

```
:CONFigure:SRWireless:BT:PLENgth
```

Query

```
:CONFigure:SRWireless:BT:PLENgth?
```

Response

#	Description	Format
1	*Bluetooth* (BR & EDR) payload length	Integer

Parameters

#	Description	Format	Units	Min	Max	Default
1	*Bluetooth* (BR & EDR) payload length	Number+	---	0	See notes	27

Note:

The maximum payload length depends on the packet type according to the following table.

DH1	27
DH3	183
DH5	339
2DH1	54
2DH3	367
2DH5	679
3DH1	83
3DH3	552
3DH5	1021

Example of Use

```
:CONF:SRW:BT:PLEN 1021      // Set the payload length (assuming
                            // a 3DH5 packet).
```

:CONFigure:SRWireless:BT:PPOWer:ULIMit

Function

Set or query the *Bluetooth* peak power upper limit (TRM/CA/01/C).

Command

```
:CONFigure:SRWireless:BT:PPOWer:ULIMit
```

Query

```
:CONFigure:SRWireless:BT:PPOWer:ULIMit?
```

Response

#	Description	Format	Units
1	Peak power upper limit	Real number	dBm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Peak power upper limit	Number+	dBm	–100	100	23	0.01 dB

Example of Use

```
:CONF:SRW:BT:PPOW:ULIM 22           // Set limit to 22 dBm
```

:CONFigure:SRWireless:BT:RDEVm:ULIMit

Function

Set or query the *Bluetooth* EDR RMS DEVM upper limit (TRM/CA/11/C).

Command

```
:CONFigure:SRWireless:BT:RDEVm:ULIMit
```

Query

```
:CONFigure:SRWireless:BT:RDEVm:ULIMit?
```

Response

#	Description	Format	Units
1	EDR RMS DEVM upper limit – π/4-DQPSK blocks	Real number	---
2	EDR RMS DEVM upper limit – 8DPSK blocks	Real number	---

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	EDR RMS DEVM upper limit – π/4-DQPSK blocks	Number+		0	1	0.2	0.001
2	EDR RMS DEVM upper limit – 8DPSK blocks	Number+		0	1	0.13	0.001

Example of Use

```
:CONF:SRW:BT:RDEV:ULIM 0.2,0.13
```

:CONFigure:SRWireless:DCONversion

Function

Set or query the instrument receiver downconversion method for this segment.
This function allows control of the downconversion process within the instrument's receiver. It allows automatic or manual control of the local oscillator (LO) frequency, with automatic control being the default setting.
The setting affects whether the LO frequency is below the RF input frequency (low side injection) or above the RF input frequency (high side injection).
In AUTOMATIC mode the choice of low or high side injection is selected automatically according to the measurement frequency setting in order to achieve the optimum noise performance on spectrum measurements.
Under certain circumstances it may be desirable to override the automatic setting and use the LO_LOWER and LO_UPPER settings to force low side injection or high side injection respectively. These settings could be of use to advanced users in cases where the default LO injection method results in image signals appearing at the IF (intermediate frequency) and measurement performance is affected.
This setting has no effect on CW, WLAN 80+80 and 160 MHz full span spectrum measurements. For these cases the LO injection method is always determined automatically.

Command

```
:CONFigure:SRWireless:DCONversion
```

Query

```
:CONFigure:SRWireless:DCONversion?
```

Response

#	Description	Format
1	Downconversion method	See below

Parameters

#	Description	Format	Units	Min	Max	Default
1	Downconversion method	See below	---	---	---	AUTOMATIC

Downconversion method:

AUTOMATIC	The LO injection method is determined automatically.
LO_LOWER	Use low side LO injection (LO < RF)
LO_UPPER	Use high side LO injection (LO > RF)

Example of Use

```
:CONF:SRW:DCON LO_LOWER     // Use low side LO injection.
```

:CONFigure:SRWireless:CAPTure:MODE

Function

Set or query the capture mode for the active segment.

If the capture mode is set to TIME, the system will always capture for that time. If the mode is set to PACKET, the system will count packets and terminate the capture segment as soon as the required number has been received.

Note:
The capture will terminate when the capture time expires regardless of the number of packets counted. The capture time, therefore, acts as an upper limit on the capture time when the capture mode is PACKET.

Command

```
:CONFigure:SRWireless:CAPTure:MODE
```

Query

```
:CONFigure:SRWireless:CAPTure:MODE?
```

Response

#	Description	Format
1	Capture mode	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	Capture mode	See definition	---	---	---	TIME

Capture mode:

TIME	The system will capture all packets received within the specified capture time.
PACKET	The system will count packets as they are received and terminate the capture segment when at least that number has been counted or the capture time has expired.

Example of Use

```
:CONF:SRW:TIME 100E-03        // Set the upper limit for capture time to 100ms.
:CONF:SRW:CAPT:MODE PACKET    // Set the capture mode to count packets
:CONF:SRW:PACK 10             // Set the number of packets required.
                              // Capture will stop after 10 packets have
                              // been counted, or the time exceeds 100 ms.
```

:CONFigure:SRWireless:CAPTure:TIME?

Function

Query only.

Returns total available capture time (i.e. h/w limit) and available free time.

Command

N/A

Query

```
:CONFigure:SRWireless:CAPTure:TIME?
```

Response

#	Description	Format	Units
1	Total time available from hardware	Real number	s
2	Available free time	Real number	s

Parameters

N/A

Example of Use

```
:CONF:SRW:CAPT:TIME?
> 1.24, 1.23
```

:CONFigure:SRWireless:CW:FREQuency:LIMit

Function

Set or query the CW frequency offset limit (±).

Command

```
:CONFigure:SRWireless:CW:FREQuency:LIMit
```

Query

```
:CONFigure:SRWireless:CW:FREQuency:LIMit?
```

Response

#	Description	Format	Units
1	CW frequency offset limit (±)	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	CW frequency offset limit (±)	Number+	Hz	0	1 MHz	1 MHz	1 Hz

Example of Use

```
:CONF:SRW:CW:FREQ:LIM 200000// Set limit to ±200 kHz.
```

:CONFigure:SRWireless:CW:POWer:LLIMit

Function

Set or query the CW power lower limit.

Command

```
:CONFigure:SRWireless:CW:POWer:LLIMit
```

Query

```
:CONFigure:SRWireless:CW:POWer:LLIMit?
```

Response

#	Description	Format	Units
1	CW power lower limit	Real number	dBm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	CW power lower limit	Number+	dBm	–100.00	+100.00	–100.00	0.01 dB

Note:

Default value ensures that all measurements will pass.

Example of Use

```
:CONF:SRW:CW:POW:LLIM -34.56
```

:CONFigure:SRWireless:CW:POWer:ULIMit

Function

Set or query the CW power upper limit.

Command

```
:CONFigure:SRWireless:CW:POWer:ULIMit
```

Query

```
:CONFigure:SRWireless:CW:POWer:ULIMit?
```

Response

#	Description	Format	Units
1	CW power upper limit	Real number	dBm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	CW power upper limit	Number+	dBm	–100.00	100.00	100.00	0.01 dB

Note:

Default value ensures that all measurements will pass.

Example of Use

```
:CONF:SRW:CW:POW:ULIM -12.34
```

:CONFigure:SRWireless:DEViation:PPSYmbol

Function

Set the number of points per symbol returned when retrieving deviation v symbol measurements (*Bluetooth*). The higher the number, the more detailed the resultant graph when plotted, although the amount of data to transfer will be larger.

Command

```
:CONFigure:SRWireless:DEViation:PPSYmbol
```

Query

```
:CONFigure:SRWireless:DEViation:PPSYmbol?
```

Response

#	Description	Format
1	Points per symbol	Integer

Parameters

#	Description	Format	Units	Min	Max	Default
1	Points per symbol	Number+	---	2	32	8

Example of Use

```
:CONF:SRW:DEV:PPSY 4
```

:CONFigure:SRWireless:DSSS:ALENgth

Function

Set or query the DSSS EVM analysis length in chips. (For use when the analysis length mode is USER)

Command

```
:CONFigure:SRWireless:DSSS:ALENgth
```

Query

```
:CONFigure:SRWireless:DSSS:ALENgth?
```

Response

#	Description	Format
1	DSSS EVM analysis length	Integer

Parameters

#	Description	Format	Units	Min	Max	Default
1	DSSS EVM analysis length	Number+	---	1	11000	1000

Example of Use

```
:CONF:SRW:DSSS:ALM USER      // Set user mode.
:CONF:SRW:DSSS:ALEN 2000     // Set the analysis length.
```

:CONFigure:SRWireless:DSSS:ALMode

Function

Set or query the DSSS EVM analysis length mode.

Command

```
:CONFigure:SRWireless:DSSS:ALMode
```

Query

```
:CONFigure:SRWireless:DSSS:ALMode?
```

Response

#	Description	Format
1	DSSS EVM analysis length mode.	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	DSSS EVM analysis length mode	See definition	---	---	---	USER

DSSS EVM analysis length mode:

USER	Analysis length is user defined. (Use `:CONF:SRW:DSSS:ALEN` to set the analysis length in chips.)
FULLPACKET	Analysis performed on full packet.

Example of Use

```
:CONF:SRW:DSSS:ALM USER       // Set user mode.
:CONF:SRW:DSSS:ALEN 2000      // Set the analysis length.
```

:CONFigure:SRWireless:DSSS:CCTLength

Function

Set or query the DSSS chip clock frequency tolerance analysis length in chips.

Command

```
:CONFigure:SRWireless:DSSS:CCTLength
```

Query

```
:CONFigure:SRWireless:DSSS:CCTLength?
```

Response

#	Description	Format
1	DSSS chip clock tolerance analysis length	Integer

Parameters

#	Description	Format	Units	Min	Max	Default
1	DSSS chip clock frequency tolerance analysis length	Number+	---	1	30250	1000

Example of Use

```
:CONF:SRW:DSSS:CCTL MAX      // Set analysis length to maximum.
```

5

Remote Operation

:CONFigure:SRWireless:DSSS:CMEThod

Function

Set or query the DSSS EVM calculation method.

When set to USER the EVM measurement is calculated using chips transmitted during the packet payload. When set to PREAMBLE_HDR, the measurement is performed using the definition in IEEE 802.11b-1999 (18.4.7.8) and is calculated over 1000 chips that are transmitted during the PLCP preamble and header.

Command

```
:CONFigure:SRWireless:DSSS:CMEThod
```

Query

```
:CONFigure:SRWireless:DSSS:CMEThod?
```

Response

#	Description	Format
1	WLAN DSSS EVM calculation method	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN DSSS EVM calculation method	See definition	---	---	---	USER

WLAN DSSS EVM calculation method:
USER
PREAMBLE_HDR

Example of Use

```
:CONF:SRW:DSSS:CMET PREAMBLE_HDR
```

:CONFigure:SRWireless:FREQuency

Function

Set or query the RX frequency for the active segment.

Command

```
:CONFigure:SRWireless:FREQuency
```

Query

```
:CONFigure:SRWireless:FREQuency?
```

Response

#	Description	Format	Units
1	Receiver frequency for active segment	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Receiver frequency for active segment	Number+	Hz	See notes	See notes	See notes	1 Hz

Note:

The short range wireless application limits the minimum frequency to 50 MHz when there is MX887032A licence installed, otherwise frequency is limited to 1.9 GHz. The minimum frequency compatible with the 200 MHz sampling rate is 1.9 GHz. The sampling rate used below 1.9 GHz is 100 MHz.

The maximum frequency is 6 GHz if the MU887000A-001 6 GHz Frequency Extension option is fitted. Otherwise it is 3.8 GHz.

Default frequencies depend on the wireless standard / segment type:

CW, 802.11b, 802.11g:	2412 MHz
802.11a, 802.11n, 802.11ac	4920 MHz (2412 MHz if 6 GHz extension is not fitted)
802.11ax	
802.11p	5850 MHz
BT / BLE	2402 MHz
AUTOBT	2402 MHz
AUTODSSS	2412 MHz
AUTOOFDM	2412 MHz
COMPMIMO	4920 MHz (2412 MHz if 6 GHz extension is not fitted)
SEQMIMO	4920 MHz (2412 MHz if 6 GHz extension is not fitted)
TMIMOM	2412 MHz
TMIMOS	2412 MHz

SEQ80_80SISO	5200 MHz (first segment) / 5500 MHz (second segment)
CON80_80SISOM	5200 MHz
CON80_80SISOS	5500 MHz
CON80_80TMIMOM	5200 MHz
CON80_80TMIMOS	Segments alternate 5500 MHz / 5200 MHz
SEQ80_80TMIMOM	5200 MHz (first segment) / 5500 MHz (second segment)
SEQ80_80TMIMOS	Segments alternate 5200 MHz / 5500 MHz

Example of Use

```
:CONF:SRW:FREQ 2412MHz
```

:CONFigure:SRWireless:GATE:ALL

Function

Command form enables or disables all gates at once. The query form reads the current status of all gates. (Gates may be enabled and disabled individually with `:CONF:SRW:GATE:ENAB`.)

Note:

Gate 1 cannot be disabled and is permanently set to "packet". This command is not valid for CW or *Bluetooth* segments.

Command

```
:CONFigure:SRWireless:GATE:ALL
```

Query

```
:CONFigure:SRWireless:GATE:ALL?
```

Response

#	Description	Format
1	Gate 1 enabled	Boolean
2	Gate 2 enabled	Boolean
3	Gate 3 enabled	Boolean
4	Gate 4 enabled	Boolean
5	Gate 5 enabled	Boolean
6	Gate 6 enabled	Boolean
7	Gate 7 enabled	Boolean
8	Gate 8 enabled	Boolean

Parameters

#	Description	Format
1	Enable or disable all gates. ('1' or 'ON' means enable.) (Not required for query.)	Boolean

Example of Use

When a segment is created all gates are disabled. To enable more than 4 gates it is convenient to use this command to enable all gates, and then selectively disable the gates that are not required.

```
:CONF:SRW:GATE:ALL ON         // Enable all gates
:CONF:SRW:GATE:ENAB 8, OFF    // Disable gate 8
:CONF:SRW:GATE:ENAB 7, OFF    // Disable gate 7

:CONF:SRW:GATE:ALL?           // Check the status of the gates.
> 1,1,1,1,1,1,0,0
```

5

Remote Operation

:CONFigure:SRWireless:GATE:ENABle

Function

Enable / Disable individual gates.

Note:

Gate 1 cannot be disabled and is permanently set to "packet". This command is not valid for CW or *Bluetooth* segments.

Command

```
:CONFigure:SRWireless:GATE:ENABle
```

Query

```
:CONFigure:SRWireless:GATE:ENABle?
```

Response

#	Description	Format
1	Gate enabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Gate number	Number	---	1	8	---
2	Gate state. (Not required for query)	Boolean	---	---	---	---

Example of Use

```
:CONF:SRW:GATE:ENAB 3,ON     // Enable gate 3
:CONF:SRW:GATE:ENAB? 1       // Query gate 1 (always enabled)
> 1
```

:CONFigure:SRWireless:GATE:STARt

Function

Set or query the gate start value relative to the start of the packet (when gate type = USER). The gate start value may be negative to allow the gate to include part of the capture before the start of the packet.

Note:

Gate 1 cannot be disabled and is permanently set to "packet". This command is not valid for CW or *Bluetooth* segments.

Command

```
:CONFigure:SRWireless:GATE:STARt
```

Query

```
:CONFigure:SRWireless:GATE:STARt?
```

Response

#	Description	Format	Units
1	Gate start time relative to start of packet	Real number	s

Gate start time relative to start of packet

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Gate number	Number	---	2	8	2	---
2	Gate start time relative to the start of a packet. (Not required for query.)	Number+	s	–10 ms	200 ms	0 s	1 µs

Example of Use

```
:CONF:SRW:GATE:STAR 2, 100us        // Set gate 2 start to 100µs
:CONF:SRW:GATE:STAR? 2              // Query gate 2
> 0.0001
```

:CONFigure:SRWireless:GATE:TYPE

Function

Set or query the gate type.

Note:

Gate 1 cannot be disabled and is permanently set to "packet". This command is not valid for CW or *Bluetooth* segments.

Command

```
:CONFigure:SRWireless:GATE:TYPE
```

Query

```
:CONFigure:SRWireless:GATE:TYPE?
```

Response

#	Description	Format
1	Gate type	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	Gate number	Number	---	2	8	2
2	Gate type. (Not required for query.)	See definition	---	---	---	PACKET

Gate type:

USER	The gate is defined by user-set start time and width.
PACKET	The gate is adjusted automatically to encompass the whole packet.

Example of Use

```
:CONF:SRW:GATE:TYPE 2,USER          // Set gate 2 type to USER.
:CONF:SRW:GATE:TYPE? 2              // Query gate 2
> USER
```

:CONFigure:SRWireless:GATE:WIDTh

Function

Set or query the gate width value (when gate type is USER).

Note:

Gate 1 cannot be disabled and is permanently set to “packet”. This command is not valid for CW or *Bluetooth* segments.

Command

```
:CONFigure:SRWireless:GATE:WIDTh
```

Query

```
:CONFigure:SRWireless:GATE:WIDTh?
```

Response

#	Description	Format	Units
1	Gate width	Real number	s

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Gate number	Number	---	2	8	2	---
2	Gate width. (Not require for query.)	Number+	s	0 s	200 ms	50 µs	1 µs

Example of Use

```
:CONF:SRW:GATE:WIDT 2,100us          // Set width of gate 2 to 100 µs.
:CONF:SRW:GATE:WIDT? 2               // Query the width
> 0.0001
```

:CONFigure:SRWireless:IQ:INFormation:WRITe

Function

Write information about a capture to the instrument. (Used in conjunction with `CONF:SRW:IQ:LOAD`).

Command

`:CONFigure:SRWireless:IQ:INFormation:WRITe`

Query

N/A

Response

N/A

Parameters

#	Description	Format
1	Segment number	Number
2	Error code (from original capture). Normally this value will be zero	Number
3	Segment offset in IQ sample pairs from the start of the capture.	Number
4	Segment size in IQ sample pairs (including any settling time).	Number
5	[Optional] Segment offset from start of capture to usable data in IQ sample pairs. (Skips settling time.)	Number
6	[Optional] Segment size in IQ sample pairs (excluding settling time)	Number
7	[Optional] Segment sample rate in MS/s. (Allowed values are 100 and 200. 200 MS/s is assumed if the parameter is omitted.)	Number

Example of Use

```
:CONF:SRW:IQ:INF:WRIT 1,0,0,2000000
```

:CONFigure:SRWireless:IQ:LOAD[:BINary]

Function

Upload binary IQ data to the instrument. (Used for diagnostic purposes in conjunction with `:CONF:SRW:IQ:INF:WRIT` and `:CALC:SRW`.)

Command

```
:CONFigure:SRWireless:IQ:LOAD[:BINary]
```

Query

N/A

Response

N/A

Parameters

#	Description	Format
1	#nmmm...mmm	Binary Header
2	<binary_data>	See Note

Note:

#	Binary data marker
n	ASCII digit specifying the length of the data length field.
mm...mm	'n' ASCII digits specifying the length of the binary data.
<binary_data>	<offset>,<size>,<iq_data>
<offset>	A string of ASCII digits representing the position in IQ sample pairs of the data with respect to the start of the capture buffer.
<size>	A string of ASCII digits representing the size of the IQ data in IQ sample pairs.
<iq_data>	I and Q values (4 bytes each) sent in the order I, Q

Example of Use

```
:CONF:SRW:IQ:LOAD #6123456<binary_data>
```

:CONFigure:SRWireless:OBW:POWer

Function

Set or query the percentage of transmitter power for which the occupied bandwidth (OBW) is calculated.

Command

```
:CONFigure:SRWireless:OBW:POWer
```

Query

```
:CONFigure:SRWireless:OBW:POWer?
```

Response

#	Description	Format	Units
1	Percentage of transmitter power for which the OBW will be determined.	Real number	%

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1		Number+	%	1	100	99	1%

Example of Use

```
:CONF:SRW:OBW:POW 90
```

:CONFigure:SRWireless:OFDM:ALENgth

Function

Set or query the OFDM EVM analysis length in symbols (used when OFDM analysis length mode is USER).

Command

```
:CONFigure:SRWireless:OFDM:ALENgth
```

Query

```
:CONFigure:SRWireless:OFDM:ALENgth?
```

Response

#	Description	Format
1	OFDM EVM analysis length	Integer

Parameters

#	Description	Format	Units	Min	Max	Default
1	OFDM EVM analysis length	Number+	---	16	2000	40

Example of Use

```
:CONF:SRW:OFDM:ALM USER      // Set the mode to USER.
:CONF:SRW:OFDM:ALEN 20       // Set the number of symbols.
```

:CONFigure:SRWireless:OFDM:ALMode

Function

Set or query the OFDM EVM analysis length mode in symbols.

Command

```
:CONFigure:SRWireless:OFDM:ALMode
```

Query

```
:CONFigure:SRWireless:OFDM:ALMode?
```

Response

#	Description	Format
1	OFDM EVM analysis length mode	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	OFDM EVM analysis length mode	See definition	---	---	---	FULLPACKET

OFDM EVM analysis length mode:

USER	Analysis length is user defined. (Use `:CONF:SRW:OFDM:ALEN` to set the analysis length in symbols.)
FULLPACKET	Analysis performed on full packet.

Example of Use

```
:CONF:SRW:OFDM:ALM USER      // Set user mode.
:CONF:SRW:OFDM:ALEN 2000     // Set the analysis length.
```

:CONFigure:SRWireless:OFDM:CEALength

Function

Set or query the OFDM clock error analysis length in symbols (for use when the OFDM clock error analysis length mode is USER).

Command

```
:CONFigure:SRWireless:OFDM:CEALength
```

Query

```
:CONFigure:SRWireless:OFDM:CEALength?
```

Response

#	Description	Format
1	OFDM clock error analysis length	Integer

Parameters

#	Description	Format	Units	Min	Max	Default
1	OFDM clock error analysis length	Number+	---	16	2000	40

Example of Use

```
:CONF:SRW:OFDM:CEAM USER     // Set the analysis length mode to USER.
:CONF:SRW:OFDM:CEAL 20       // Set the required length.
```

:CONFigure:SRWireless:OFDM:CEAMode

Function

Set or query the OFDM clock error analysis length mode.

Command

```
:CONFigure:SRWireless:OFDM:CEAMode
```

Query

```
:CONFigure:SRWireless:OFDM:CEAMode?
```

Response

#	Description	Format
1	OFDM clock error analysis length mode	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	OFDM clock error analysis length mode	See definition	---	---	---	FULLPACKET

OFDM clock error analysis length mode:

USER	Analysis length is user defined. (Use `:CONF:SRW:OFDM:CEAL` to set the analysis length in symbols.)
FULLPACKET	Analysis performed on full packet.

Example of Use

```
:CONF:SRW:OFDM:CEAM USER
```

:CONFigure:SRWireless:OFDM:CESTimation

Function

Set or query the OFDM channel estimation method.

Command

```
:CONFigure:SRWireless:OFDM:CESTimation
```

Query

```
:CONFigure:SRWireless:OFDM:CESTimation?
```

Response

#	Description	Format
1	OFDM channel estimation method	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	OFDM channel estimation method	See definition	---	---	---	LTSEQUENCE

OFDM channel estimation method:

LTSEQUENCE	Estimation uses the long training sequence.
FULLPACKET	Estimation uses the full packet.

Example of Use

```
:CONF:SRW:OFDM:CEST FULLPACKET
```

:CONFigure:SRWireless:OFDM:FCORrection

Function

Set the OFDM frequency correction method.

Command

```
:CONFigure:SRWireless:OFDM:FCORrection
```

Query

```
:CONFigure:SRWireless:OFDM:FCORrection?
```

Response

#	Description	Format
1	OFDM frequency correction method	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	OFDM frequency correction method	See definition	---	---	---	LTF

OFDM frequency correction method:

STF	Use the short training field only.
LTF	Use the short and long training fields. (This is the default case.)
SIG	Use the short and long training fields and the SIG field.
DATA	Use all the above fields and the packet data.

Example of Use

```
:CONF:SRW:OFDM:FCOR DATA
```

:CONFigure:SRWireless:OFDM:MIMO:EVM:CORRection

Function

Enable / disable MIMO enhanced EVM correction. When enabled this function corrects for the effect of cross-stream leakage within a DUT's transmitter.

Command

```
:CONFigure:SRWireless:OFDM:MIMO:EVM:CORRection
```

Query

```
:CONFigure:SRWireless:OFDM:MIMO:EVM:CORRection?
```

Response

#	Description	Format	Units
1	Setting enabled / disabled	Boolean	---

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Setting enabled / disabled	Boolean	---	---	---	ON	---

Example of Use

```
:CONF:SRW:OFDM:MIMO:EVM:CORR OFF
```

:CONFigure:SRWireless:OFDM:PTRacking

Function

Set or query the OFDM pilot tracking method.

Command

```
:CONFigure:SRWireless:OFDM:PTRacking
```

Query

```
:CONFigure:SRWireless:OFDM:PTRacking?
```

Response

#	Description	Format
1	OFDM pilot tracking method	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	OFDM pilot tracking method	See definition	---	---	---	PHASE

OFDM pilot tracking method:

PHASE	Phase tracking.
PAMPLITUDE	Phase and amplitude tracking.

Example of Use

```
:CONF:SRW:OFDM:PTR PAMPLITUDE
```

:CONFigure:SRWireless:PACKets

Function

Set or query the number of packets to measure for the active segment.

Command

```
:CONFigure:SRWireless:PACKets
```

Query

```
:CONFigure:SRWireless:PACKets?
```

Response

#	Description	Format
1	Number of packets to measure	Integer

Parameters

#	Description	Format	Units	Min	Max	Default
1	Number of packets	Number+	---	1	1000	1

Example of Use

```
:CONF:SRW:PACK 5
```

:CONFigure:SRWireless:PACKets:SKIP

Function

Sets the number of packets to skip at the start of a segment. The skipped packets are not measured. The remaining packets, (up to the number specified using :CONFigure:SRW:PACKets) are measured.

Notes:

1. When using timed captures (`:CONFigure:SRWireless:CAPTure:MODE TIME`), packet skip command has no effect. Instead, settling time can be used to give the same effect. See note #3 below.
2. When using packet counting (`:CONFigure:SRWireless:CAPTure:MODE PACKET`), the system automatically captures and ignores the packets to skip before counting the packets to measure.
3. This function has a similar purpose to `:CONFigure:SRWireless:STIMe`, except that the settling time is defined by packets rather than time. The two settings are applied one after the other, with the settling time applied first.

Command

```
:CONFigure:SRWireless:PACKets:SKIP
```

Query

```
:CONFigure:SRWireless:PACKets:SKIP?
```

Response

#	Description	Format
1	Number of packets to skip	Integer

Parameters

#	Description	Format	Units	Min	Max	Default
1	Number of packets to skip	Number+	---	0	1000	0

Example of Use

```
:CONF:SRW:PACK 10          // Set the number of packets to measure.
:CONF:SRW:PACK:SKIP 3      // Set the number of packets to skip.
```

:CONFigure:SRWireless:PDETector:MGAP

Function

Adjust the minimum inter-packet gap recognised by the packet detector.

The measurement system incorporates a packet detector that recognises rapid changes in input power level. To detect packets reliably, the packet detector must distinguish between inter-packet gaps and short periods within a packet where the power level is reduced due to modulation effects. This setting adjusts the minimum duration in microseconds that the packet detector will use to determine that a complete packet has been received.

If the minimum gap is set too short, the system may split packets into fragments and may not be able to analyse them correctly. If it is set too long, the packet detector may see the start of the next packet before it has determined that the previous packet has ended. As a result the system will be unable able to analyse any packets.

Command

```
:CONFigure:SRWireless:PDETector:MGAP
```

Query

```
:CONFigure:SRWireless:PDETector:MGAP?
```

Response

#	Description	Format	Units	Resolution
1	Minimum gap	Real number	s	1 µs

Parameters

#	Description	Format	Units	Min	Max	Def	Resolution
1	Minimum gap	Number+	s	3 µs	10 µs	5 µs	1 µs

Example of Use

```
:CONF:SRW:PDET:MGAP 6E-06    // Set the minimum gap recognised
                             // by the packet detector to 6 µs.
```

5

Remote Operation

:CONFigure:SRWireless:PDETector:MLENgth

Function

Sets the minimum length of packet that will be recognized by the system.

Notes:

This command can be used to prevent packet fragments from being analysed by the system. A typical use is when measuring Bluetooth EDR (extended data rate) packets. These packets include a guard interval of approximately 5 μs during which the RF output power of the DUT is not specified. If the RF is below the trigger level during this time and the trigger happens to be armed at that point, the system may capture the fragment of the packet after the guard interval. The packet fragment will not contribute to the measurement summary, but the number of packets in the segment will be one less than expected. This command provides a way to ignore short packets so that they are not counted by the system.

Command

```
:CONFigure:SRWireless:PDETector:MLENgth
```

Query

```
:CONFigure:SRWireless:PDETector:MLENgth?
```

Response

#	Description	Format	Units	Resolution
1	Minimum length of packet recognized by the system	Real number	s	1 μs

Parameters

#	Description	Format	Units	Min	Max	Default
1	Minimum length of packet recognized by the system	Number+	s	20 μs	200 μs	See note

Note:

The default value depends on the segment type.
Bluetooth segments: 136 μs.
WLAN segments: 20 μs.

Example of Use

```
:CONF:SRW:PDET:MLEN 180E-06 // Set the minimum packet length to 180 μs.
```

:CONFigure:SRWireless:PDRamp:ULIMit

Function

Set or query the WLAN power down ramp time upper limit.

Command

```
:CONFigure:SRWireless:PDRamp:ULIMit
```

Query

```
:CONFigure:SRWireless:PDRamp:ULIMit?
```

Response

#	Description	Format	Units
1	Power down ramp time upper limit	Real number	s

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Power down ramp time upper limit	Number+	s	0 s	10 ms	2 µs	0.01 µs

Example of Use

```
:CONF:SRW:PDR:ULIM 2.5E-06
```

:CONFigure:SRWireless:PDTHreshold

Function

Set or query the packet detector threshold level (relative to the input power level).

Command

```
:CONFigure:SRWireless:PDTHreshold
```

Query

```
:CONFigure:SRWireless:PDTHreshold?
```

Response

#	Description	Format	Units
1	Packet detector threshold	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Packet detector threshold	Number+	dB	–60	0	–20	1 dB

Example of Use

```
:CONF:SRW:PDTH -15
```

:CONFigure:SRWireless:PORamp:ULIMit

Function

Set or query the WLAN power on ramp time upper limit.

Command

```
:CONFigure:SRWireless:PORamp:ULIMit
```

Query

```
:CONFigure:SRWireless:PORamp:ULIMit?
```

Response

#	Description	Format	Units
1	Power on ramp time upper limit	Real number	s

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Power on ramp time upper limit	Number+	s	0 s	10 ms	2 µs	0.01 µs

Example of Use

```
:CONF:SRW:POR:ULIM 2.5E-06
```

:CONFigure:SRWireless:POWer

Function

Set or query the expected power level for the active segment.

Command

```
:CONFigure:SRWireless:POWer
```

Query

```
:CONFigure:SRWireless:POWer?
```

Response

#	Description	Format	Units
1	Expected power level for the active segment.	Real number	dBm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Expected power level for the active segment.	Number+	dBm	–65.0	+35.0	–10.0	1 dB

Example of Use

```
:CONF:SRW:POW -15            // Set the expected power level to -15 dBm.
```

:CONFigure:SRWireless:POWer:LLIMit

Function

Set or query the WLAN Transmit power lower limit.

Command

```
:CONFigure:SRWireless:POWer:LLIMit
```

Query

```
:CONFigure:SRWireless:POWer:LLIMit?
```

Response

#	Description	Format	Units
1	Transmitter average power lower limit	Real number	dBm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Transmitter average power lower limit	Number+	dBm	–100	100	–10	0.01 dB

Example of Use

```
:CONF:SRW:POW:LLIM -12
```

:CONFigure:SRWireless:POWer:ULIMit

Function

Set or query the WLAN Transmit power upper limit.

Command

```
:CONFigure:SRWireless:POWer:ULIMit
```

Query

```
:CONFigure:SRWireless:POWer:ULIMit?
```

Response

#	Description	Format	Units
1	Transmitter average power upper limit	Real number	dBm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Transmitter average power upper limit	Number+	dBm	–100.00	100.00	30	0.01 dB

Example of Use

```
:CONF:SRW:POW:ULIM 10        // Set upper limit to +10 dBm
```

:CONFigure:SRWireless:PPRofile:MODE

Function

Set the power profile mode.

Command

```
:CONFigure:SRWireless:PPRofile:MODE
```

Query

```
:CONFigure:SRWireless:PPRofile:MODE?
```

Response

#	Description	Format
1	Power profile mode	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	Power profile mode	See definition	---	---	---	AUTO

Power profile mode:

AUTO	The start and stop times of the profile are determined automatically.
USER	The start and stop times of the power profile may be set by the user.

Example of Use

```
:CONF:SRW:PPR:MODE USER      // Allow user setting of
                             // start and stop times.
```


5 Remote Operation

:CONFigure:SRWireless:PPRofile:POINts

Function

Set the number of points in the power profile.

Command

```
:CONFigure:SRWireless:PPRofile:POINts
```

Query

```
:CONFigure:SRWireless:PPRofile:POINts?
```

Response

#	Description	Format
1	Number of points	Integer

Parameters

#	Description	Format	Units	Min	Max	Default
1	Number of points	Number+	---	200	8192	8192

Example of Use

```
:CONF:SRW:PPR:POIN 1024
```

:CONFigure:SRWireless:PPRofile:STARt

Function

Set the start time of the power profile - relative to the packet rising edge. (WLAN) or PO(*Bluetooth*). Negative values allow the profile to include capture data before the start of the packet.

Note that the power profile mode must be USER for this setting to have an effect; in AUTO mode the start and stop times are determined automatically.

Command

```
:CONFigure:SRWireless:PPRofile:STARt
```

Query

```
:CONFigure:SRWireless:PPRofile:STARt?
```

Response

#	Description	Format	Units
1	Start time	Real number	s

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Start time	Number+	s	–0.1 s	1.0 s	–5.0 µs	1 µs

Example of Use

```
:CONF:SRW:PPR:STAR -10us          // Set the start time
                                  // to 10 µs before the
                                  // start of the packet.
```

:CONFigure:SRWireless:PPRofile:STOP

Function

Set the stop time of the power profile - relative to the packet rising edge. (WLAN) or PO(*Bluetooth*). Negative values allow the profile to include capture data before the start of the packet.

Note that the power profile mode must be USER for this setting to have an effect; in AUTO mode the start and stop times are determined automatically.

Command

```
:CONFigure:SRWireless:PPRofile:STOP
```

Query

```
:CONFigure:SRWireless:PPRofile:STOP?
```

Response

#	Description	Format	Units
1	Stop time	Real number	s

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Stop time	Number+	s	–0.1 s	1.0 s	20.0 µs	1 µs

Example of Use

```
:CONF:SRW:PPR:STOP 50E-06
```

:CONFigure:SRWireless:SEGMent:ACTive

Function

Set or query the active segment.

Command

```
:CONFigure:SRWireless:SEGMent:ACTive
```

Query

```
:CONFigure:SRWireless:SEGMent:ACTive?
```

Response

#	Description	Format	Units
1	Active segment index	Integer	---

Parameters

#	Description	Format	Units	Min	Max	Default
1	Segment index	Number+	---	1	n	n

Note:

If no segment is defined, the returned value for this query will be 0.
"n" is the total number of segments defined. When a new segment is added, "n" changes and this new segment becomes the active segment.

Example of Use

```
:CONF:SRW:SEGM:ACT 1
```

:CONFigure:SRWireless:SEGMent:APPend

Function

Append a new segment with a specified wireless standard. If the wireless standard is set to AUTODSSS, 802.11b packets will be identified automatically. If the wireless standard is set to AUTOOFDM, any 802.11g, a, p, n or ac packets will be identified automatically.

Command

```
:CONFigure:SRWireless:SEGMent:APPend
```

Query

N/A

Response

N/A

Parameters

#	Description	Format	Units	Min	Max	Default
1	Wireless standard	See below	---	---	---	
2	Optional string to identify segment	String	---	---	---	"No name" (max 1024 bytes)

Wireless standard:

CW	CW
WLA	WLAN 802.11a
WLB	WLAN 802.11b
WLG	WLAN 802.11g
WLP	WLAN 802.11p
WLN	WLAN 802.11n
WLAC	WLAN 802.11ac
WLAX	WLAN 802.11ax
AUTODSSS	Automatically identify WLAN DSSS packets
AUTOOFDM	Automatically identify WLAN OFDM packets
BT	Bluetooth
BLE	Bluetooth Low Energy
AUTOBT	Automatically identify Bluetooth and BLE packets
COMPMIMO	WLAN Composite MIMO
SEQMIMO	WLAN Sequential MIMO
TMIMOM	WLAN True MIMO (Master “M” module)
TMIMOS	WLAN True MIMO (“S” Module control)
SEQ80_80SISO	WLAN 802.11ac sequential 80+80 SISO
CON80_80SISOM	WLAN 802.11ac concurrent 80+80 SISO (“M” module control)

CON80_80SISOS	WLAN 802.11ac concurrent 80+80 SISO (“S” module control)
CON80_80TMIMOM	WLAN 802.11ac concurrent 80+80 True MIMO (“M” module control)
CON80_80TMIMOS	WLAN 802.11ac concurrent 80+80 True MIMO (“S” module control)
SEQ80_80TMIMOM	WLAN 802.11ac sequential 80+80 True MIMO (“M” module control)
SEQ80_80TMIMOS	WLAN 802.11ac sequential 80+80 True MIMO (“S” module control)

Example of Use

```
:CONF:SRW:SEG:APP AUTOOFDM, "Any OFDM packet"
```

:CONFigure:SRWireless:SEGMent:AUToid:OFDM:OVERride:BW20

Function

Specify how 20 MHz bandwidth OFDM legacy packets will be interpreted.

The purpose of this command is to allow the user to specify how the OFDM Auto-ID system will interpret 20 MHz BW OFDM legacy packets. Its main purpose is instruct the system to interpret such packets in the 5 GHz WLAN band as 802.11p signals rather than 802.11a.

The default Auto-ID behaviour for interpreting 20 MHz legacy packets depends on the carrier frequency is as follows:

Carrier Frequency	Interpretation
Less than 1 GHz	802.11p
Between 1 GHz and 4 GHz inclusive	802.11g
Greater than 4 GHz	802.11a

The default interpretation in each of these frequency bands can be overridden by specifying a standard type.

Command

```
:CONFigure:SRWireless:SEGMent:AUToid:OFDM:OVERride:BW20
```

Query

```
:CONFigure:SRWireless:SEGMent:AUToid:OFDM:OVERride:BW20?
```

Response

#	Description	Format	Units	Resolution
1	Override standard for frequencies less than 1 GHz. (Covers 700 MHz band)	See definition	---	---
2	Override standard for frequencies between 1 GHz and 4 GHz inclusive. (Covers 2.4 GHz band.)	See definition	---	---
3	Override standard for frequencies greater than 4 GHz (Covers 5 GHz band)	See definition	---	---

Parameters

#	Description	Format	Units	Min	Max	Default
1	Override standard for frequencies less than 1 GHz. (Covers 700 MHz band)	See definition	---	---	---	AUTO
2	Override standard for frequencies between 1 GHz and 4 GHz inclusive. (Covers 2.4 GHz band.)	See definition	---	---	---	AUTO
3	Override standard for frequencies greater than 4 GHz (Covers 5 GHz band)	See definition	---	---	---	AUTO

Override standard:

AUTO	Allow the Auto-ID system to identify the standard automatically.
WLA	Interpret 20 MHz bandwidth legacy packets as 802.11a.
WLG	Interpret 20 MHz bandwidth legacy packets as 802.11g.
WLP	Interpret 20 MHz bandwidth legacy packets as 802.11p.

Example of Use

```
:CONF:SRW:SEGM:AUT:OFDM:OVER:BW20 AUTO,AUTO,WLP   // Interpret 20 MHz
                                                  // legacy signals above
                                                  // 4 GHz as 802.11p
                                                  // rather than 802.11a.
```

:CONFigure:SRWireless:SEGMent:CLEar

Function

Clear all segments.

Command

`:CONFigure:SRWireless:SEGMent:CLEar`

Query

N/A

Response

N/A

Parameters

N/A

Example of Use

`:CONF:SRW:SEGM:CLE`

:CONFigure:SRWireless:SEGMent:NUMBer?

Function

Query only. Returns number of segments defined.

Command

N/A

Query

```
:CONFigure:SRWireless:SEGMent:NUMBer?
```

Response

#	Description	Format	Units
1	Number of segments defined	Integer	---

Note:

The maximum number of segments allowed is 64.

Parameters

N/A

Example of Use

```
:CONF:SRW:SEGM:NUMB?
> 1
```

:CONFigure:SRWireless:SEGMent:PORT

Function

Sets the instrument input port to be used for the active segment. Using this command it is possible to specify a different input port for each segment of a multi-segment capture. For MIMO device testing, the antenna outputs of the DUT are connected to separate instrument ports. This command is used to associate each port with a separate capture segment, allowing the DUT spatial streams to be measured rapidly in sequence with a single capture.

By default the port associated with a segment is SYSTEM - the port that had previously been set up using `:ROUT:PORT:CONN:DIR`. The port is only changed from the SYSTEM setting if this command is used to assign a specific port, and the setting remains in force only for the duration of the segment. At the end of the segment the original SYSTEM port is restored.

Command

`:CONFigure:SRWireless:SEGMent:PORT`

Query

`:CONFigure:SRWireless:SEGMent:PORT?`

Response

#	Description	Format	Units	Res
1	Port identification	See below	---	---

Parameters

#	Description	Format	Units	Min	Max	Default
1	Port identification	See below	---	---	---	SYSTEM

Port identification:

SYSTEM	The port previously set using `:ROUT:PORT:CONN:DIR`
PORT1	Port 1
PORT2	Port 2
PORT3	Port 3
PORT4	Port 4

Example of Use

```
// Use port 3 for segment 1 and port 4 for segment 2
:CONF:SRW:SEGM:ACT 1         // Set segment 1 as the active segment.
:CONF:SRW:SEGM:PORT PORT3    // Use port 3 for segment 1.
:CONF:SRW:SEGM:ACT 2         // Set segment 2 as the active segment.
:CONF:SRW:SEGM:PORT PORT4    // Use port 4 for segment 2.
```

:CONFigure:SRWireless:SEGMent:REMove

Function

Remove a segment.

Command

```
:CONFigure:SRWireless:SEGMent:REMove
```

Query

N/A

Response

N/A

Parameters

#	Description	Format	Units	Min	Max	Default
1	Index of segment to remove	Number+	---	1	n	n

Note:

n is the number of segments. The last segment in the list – segment n - will become the active segment after a segment is removed.

Example of Use

```
:CONF:SRW:SEGM:REM 2         // Remove segment 2.
```

:CONFigure:SRWireless:SEGMent:STATe

Function

Enable / Disable the active segment. Segments are enabled by default when first created. This command allows them to be disabled. A disabled segment will not be executed when a measurement capture is initiated. At least one segment must be enabled before initiating a measurement.

This command allows advanced users to create and maintain a set of segments for different wireless standards. Individual segments may be enabled and configured before each capture while the remainder are left disabled. This is more efficient, and hence faster, than clearing and recreating a segment each time a different wireless standard is required to be tested.

Command

```
:CONFigure:SRWireless:SEGMent:STATe
```

Query

```
:CONFigure:SRWireless:SEGMent:STATe?
```

Response

#	Description	Format	Units	Res
1	Segment enable / disable flag	Boolean	---	---

Parameters

#	Description	Format	Units	Min	Max	Default
1	Segment enable / disable flag	Boolean	---	---	---	ON

Example of Use

```
// Re-enabling and modifying an existing segment is quicker than using
// CLEar followed by APPend and setting up the segment from scratch.

// Create two segments for later use.
:CONF:SRW:SEGM:CLE            // Start with no segments.
:CONF:SRW:SEGM:APP WLG        // Set up segment 1 for 802.11g measurements
:CONF:SRW:SEGM:APP WLN        // Set up segment 2 for 802.11n measurements

// Enable only the WLG segment and make measurements.
:CONF:SRW:SEGM:ACT 2          // Make the second segment (802.11n) active
:CONF:SRW:SEGM:STAT OFF       // Disable it
:CONF:SRW:SEGM:ACT 1          // Make the first segment (802.11g)active
:CONF:SRW:SEGM:STAT ON        // Enable it.
```

```
// Configure Segment 1 and initiate a measurement
// …

// Enable only the WLN segment and make measurements
:CONF:SRW:SEGM:ACT 1         // Make the first segment (802.11g) active
:CONF:SRW:SEGM:STAT OFF      // Disable it
:CONF:SRW:SEGM:ACT 2         // Make the second segment (802.11n) active
:CONF:SRW:SEGM:STAT ON       // Enable it.

// Configure Segment 2 and initiate a measurement
// …

// Re-enable only the WLG segment and make new measurements with different
// settings.
:CONF:SRW:SEGM:ACT 2         // Make the second segment (802.11n) active
:CONF:SRW:SEGM:STAT OFF      // Disable it
:CONF:SRW:SEGM:ACT 1         // Make the first segment (802.11g)active
:CONF:SRW:SEGM:STAT ON       // Enable it.

// Configure Segment 1 and initiate a measurement
// …
```

:CONFigure:SRWireless:SEGMent:TMIMo:IP

Function

Associates an MU887000A module's IP address with a TMIMOS segment.

Notes:

1. The active segment must be a TMIMOS segment type.

2. This command does not affect the IP address setting of a MU887000A module. It merely associates a TMIMOS segment with the IP address of the module that will be used to capture one of a MIMO DUT's spatial streams. This allows the controlling master module (the "M" module) to take control of other modules for the purpose of configuration and data transfer.

Command

```
:CONFigure:SRWireless:SEGMent:TMIMo:IP
```

Query

```
:CONFigure:SRWireless:SEGMent:TMIMo:IP?
```

Response

#	Description	Format
1	IP Address (dotted decimal format)	String

Parameters

#	Description	Format	Units	Min	Max	Default
1	IP Address (dotted decimal format)	Boolean	---	---	---	"0.0.0.0"

Example of Use

```
:CONF:SRW:SEGM:TMIM:IP "192.168.10.1"
```

:CONFigure:SRWireless:SELect:BLE:CDRift

Function

Enable / disable *Bluetooth* LE measurements - carrier frequency offset and drift (TRM-LE/CA/BV-06-C, TRM-LE/CA/BV-12-C and TRM-LE/CA/BV-14-C).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BLE:CDRIFT?
:FETCh:SRWireless:PACKet:BLE:CDRIFT?

Command

```
:CONFigure:SRWireless:SELect:BLE:CDRift
```

Query

```
:CONFigure:SRWireless:SELect:BLE:CDRift?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BLE:CDR ON
```

:CONFigure:SRWireless:SELect:BLE:IBEMissions

Function

Enable / disable Bluetooth Low Energy in-band emissions measurement (TRM-LE/CA/BV-03-C, TRM-LE/CA/BV-08-C).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BLE:IBEMissions?
:FETCh:SRWireless:PACKet:BLE:IBEMissions?

Command

```
:CONFigure:SRWireless:SELect:BLE:IBEMissions
```

Query

```
:CONFigure:SRWireless:SELect:BLE:IBEMissions?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BLE:IBEM ON
```

:CONFigure:SRWireless:SELect:BLE:MODulation

Function

Enable / disable *Bluetooth* LE measurements - modulation characteristics (TRM-LE/CA/BV-05-C, TRM-LE/CA/BV-09-C, TRM-LE/CA/BV-10-C, TRM-LE/CA/BV-11-C, and TRM-LE/CA/BV-13-C).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BLE:MODulation?
:FETCh:SRWireless:PACKet:BLE:MODulation?

Command

```
:CONFigure:SRWireless:SELect:BLE:MODulation
```

Query

```
:CONFigure:SRWireless:SELect:BLE:MODulation?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BLE:MOD ON
```

:CONFigure:SRWireless:SELect:BLE:POWer

Function

Enable / disable *Bluetooth* LE measurements - output power (TRM-LE/CA/BV-01-C).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BLE:POWer?
:FETCh:SRWireless:PACKet:BLE:POWer?

Command

```
:CONFigure:SRWireless:SELect:BLE:POWer
```

Query

```
:CONFigure:SRWireless:SELect:BLE:POWer?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BLE:POW ON
```

:CONFigure:SRWireless:SELect:BT:ACPower

Function

Enable / disable *Bluetooth* measurements - adjacent channel power (TRM/CA/06/C).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BT:ACPower?
:FETCh:SRWireless:PACKet:BT:ACPower?

Command

```
:CONFigure:SRWireless:SELect:BT:ACPower
```

Query

```
:CONFigure:SRWireless:SELect:BT:ACPower?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:ACP ON
```

:CONFigure:SRWireless:SELect:BT:BANDwidth

Function

Enable / disable Bluetooth 20 dB bandwidth measurement (TRM/CA/05/C).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BT:BANDwidth?
:FETCh:SRWireless:PACKet:BT:BANDwidth?

Command

```
:CONFigure:SRWireless:SELect:BT:BANDwidth
```

Query

```
:CONFigure:SRWireless:SELect:BT:BANDwidth?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:BAND ON
```

:CONFigure:SRWireless:SELect:BT:CDRift

Function

Enable / disable *Bluetooth* measurements - carrier drift (TRM/CA/09/C).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BT:CDRift?
:FETCh:SRWireless:PACKet:BT:CDRift?

Command

```
:CONFigure:SRWireless:SELect:BT:CDRift
```

Query

```
:CONFigure:SRWireless:SELect:BT:CDRift?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:CDR ON
```

:CONFigure:SRWireless:SELect:BT:CONSt

Function

Enable / disable *Bluetooth* measurements - constellation (EDR).

For details of measurements in this group see:
:FETCh:SRWireless:PACKet:BT:CONSt[:BINary]?

Command

```
:CONFigure:SRWireless:SELect:BT:CONSt
```

Query

```
:CONFigure:SRWireless:SELect:BT:CONSt?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:CONS ON
```

:CONFigure:SRWireless:SELect:BT:DEViation

Function

Enable / disable *Bluetooth* measurements - deviation vs symbol (BR / BLE).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BT:DEViation:ASCii?
:FETCh:SRWireless:SUMMary:BT:DEViation[:BINary]?
:FETCh:SRWireless:PACKet:BT:DEViation:ASCii?
:FETCh:SRWireless:PACKet:BT:DEViation[:BINary]?

Command

```
:CONFigure:SRWireless:SELect:BT:DEViation
```

Query

```
:CONFigure:SRWireless:SELect:BT:DEViation?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:DEV ON
```

:CONFigure:SRWireless:SELect:BT:DEVM

Function

Enable / disable *Bluetooth* measurements - DEVM vs symbol (EDR).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BT:DEVM:ASCii?
:FETCh:SRWireless:SUMMary:BT:DEVM[:BINary]?
:FETCh:SRWireless:PACKet:BT:DEVM:ASCii?
:FETCh:SRWireless:PACKet:BT:DEVM[:BINary]?

Command

```
:CONFigure:SRWireless:SELect:BT:DEVM
```

Query

```
:CONFigure:SRWireless:SELect:BT:DEVM?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:DEVM ON
```

:CONFigure:SRWireless:SELect:BT:ECModulation

Function

Enable / disable *Bluetooth* measurements - EDR carrier frequency stability and modulation accuracy (TRM/CA/11/C).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BT:ECModulation?
:FETCh:SRWireless:PACKet:BT:ECModulation?

Command

```
:CONFigure:SRWireless:SELect:BT:ECModulation
```

Query

```
:CONFigure:SRWireless:SELect:BT:ECModulation?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:ECM ON
```

:CONFigure:SRWireless:SELect:BT:EDPencoding

Function

Enable / disable *Bluetooth* measurements - EDR differential phase encoding measurement (TRM/CA/12/C).

Command

```
:CONFigure:SRWireless:SELect:BT:EDPencoding
```

Query

```
:CONFigure:SRWireless:SELect:BT:EDPencoding?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:EDP ON
```

:CONFigure:SRWireless:SELect:BT:ERTPower

Function

Enable / disable *Bluetooth* measurements - EDR relative TX power (TRM/CA/10/C).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BT:ERTPower?
:FETCh:SRWireless:PACKet:BT:ERTPower?

Command

```
:CONFigure:SRWireless:SELect:BT:ERTPower
```

Query

```
:CONFigure:SRWireless:SELect:BT:ERTPower?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:ERTP ON
```

:CONFigure:SRWireless:SELect:BT:EYE

Function

Enable / disable *Bluetooth* measurements - eye diagram (BR / BLE).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BT:EYE:ASCii?
:FETCh:SRWireless:SUMMary:BT:EYE[:BINary]?
:FETCh:SRWireless:PACKet:BT:EYE:ASCii?
:FETCh:SRWireless:PACKet:BT:EYE[:BINary]?

Command

```
:CONFigure:SRWireless:SELect:BT:EYE
```

Query

```
:CONFigure:SRWireless:SELect:BT:EYE?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:EYE ON
```

:CONFigure:SRWireless:SELect:BT:FRANge

Function

Enable / disable Bluetooth frequency range measurements (TRM/CA/04/C).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BT:FRANge?
:FETCh:SRWireless:PACKet:BT:FRANge?

Command

```
:CONFigure:SRWireless:SELect:BT:FRANge
```

Query

```
:CONFigure:SRWireless:SELect:BT:FRANge?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:FRAN ON
```

:CONFigure:SRWireless:SELect:BT:IBSPurious

Function

Enable / disable *Bluetooth* measurements - EDR in band spurious emissions (TRM/CA/13/C).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BT:IBSPurious?
:FETCh:SRWireless:PACKet:BT:IBSPurious?

Command

```
:CONFigure:SRWireless:SELect:BT:IBSPurious
```

Query

```
:CONFigure:SRWireless:SELect:BT:IBSPurious?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:IBSP ON
```

:CONFigure:SRWireless:SELect:BT:ICFTolerance

Function

Enable / disable *Bluetooth* measurements - initial carrier frequency tolerance (TRM/CA/08/C).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BT:ICFTolerance?
:FETCh:SRWireless:PACKet:BT:ICFTolerance?

Command

```
:CONFigure:SRWireless:SELect:BT:ICFTolerance
```

Query

```
:CONFigure:SRWireless:SELect:BT:ICFTolerance?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:ICFT ON
```

:CONFigure:SRWireless:SELect:BT:MODulation

Function

Enable / disable *Bluetooth* measurements - modulation characteristics (TRM/CA/07/C).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BT:MODulation?
:FETCh:SRWireless:PACKet:BT:MODulation?

Command

```
:CONFigure:SRWireless:SELect:BT:MODulation
```

Query

```
:CONFigure:SRWireless:SELect:BT:MODulation?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:MOD ON
```

:CONFigure:SRWireless:SELect:BT:POWer

Function

Enable / disable *Bluetooth* measurements - transmit power (TRM/CA/01/C).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BT:POWer?
:FETCh:SRWireless:PACKet:BT:POWer?

Command

```
:CONFigure:SRWireless:SELect:BT:POWer
```

Query

```
:CONFigure:SRWireless:SELect:BT:POWer?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:POW ON
```

:CONFigure:SRWireless:SELect:BT:PPRofile

Function

Enable / disable *Bluetooth* measurements - power burst profile (BR / EDR / BLE).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BT:PPRofile:AVERage:ASCii?
:FETCh:SRWireless:SUMMary:BT:PPRofile:AVERage[:BINary]?
:FETCh:SRWireless:SUMMary:BT:PPRofile:PEAK:ASCii?
:FETCh:SRWireless:SUMMary:BT:PPRofile:PEAK[:BINary]?
:FETCh:SRWireless:PACKet:BT:PPRofile:AVERage:ASCii?
:FETCh:SRWireless:PACKet:BT:PPRofile:AVERage[:BINary]?
:FETCh:SRWireless:PACKet:BT:PPRofile:PEAK:ASCii?
:FETCh:SRWireless:PACKet:BT:PPRofile:PEAK[:BINary]?

Command

```
:CONFigure:SRWireless:SELect:BT:PPRofile
```

Query

```
:CONFigure:SRWireless:SELect:BT:PPRofile?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:PPR ON
```

:CONFigure:SRWireless:SELect:BT:SPECtrum

Function

Enable / disable *Bluetooth* measurements - spectrum (BR / EDR / BLE).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:BT:SPECtrum:ASCii?
:FETCh:SRWireless:SUMMary:BT:SPECtrum [:BINary]?
:FETCh:SRWireless:PACKet:BT:SPECtrum:ASCii?
:FETCh:SRWireless:PACKet:BT:SPECtrum[:BINary]?

Command

```
:CONFigure:SRWireless:SELect:BT:SPECtrum
```

Query

```
:CONFigure:SRWireless:SELect:BT:SPECtrum?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:SPEC ON
```

:CONFigure:SRWireless:SELect:BT:VECTor

Function

Enable / disable *Bluetooth* measurements - vector diagram (EDR).

For details of measurements in this group see:
:FETCh:SRWireless:PACKet:BT:VECTor[:BINary]?

Command

```
:CONFigure:SRWireless:SELect:BT:VECTor
```

Query

```
:CONFigure:SRWireless:SELect:BT:VECTor?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:BT:VECT ON
```

:CONFigure:SRWireless:SELect:CW:FREQuency

Function

Enable / disable CW frequency measurement.

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:CW:FREQuency?

Command

```
:CONFigure:SRWireless:SELect:CW:FREQuency
```

Query

```
:CONFigure:SRWireless:SELect:CW:FREQuency?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:CW:FREQ ON     // Enable measurement of CW frequency.
```

:CONFigure:SRWireless:SELect:CW:POWer

Function

Enable / disable CW power (average and peak).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:CW:POWer?

Command

```
:CONFigure:SRWireless:SELect:CW:POWer
```

Query

```
:CONFigure:SRWireless:SELect:CW:POWer?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:CW:POW ON
```

:CONFigure:SRWireless:SELect:CW:SPECtrum

Function

Enable / disable CW spectrum measurements.

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:CW:SPECtrum[:BINary]?
:FETCh:SRWireless:SUMMary:CW:SPECtrum:ASCii?

Command

```
:CONFigure:SRWireless:SELect:CW:SPECtrum
```

Query

```
:CONFigure:SRWireless:SELect:CW:SPECtrum?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:CW:SPEC ON
```

:CONFigure:SRWireless:SELect:WLAN:CCDF

Function

Enable / disable CCDF profile (graphical) measurements.

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:WLAN:CCDF[:BINary]?
:FETCh:SRWireless:PACKet:WLAN:CCDF[:BINary]?
:FETCh:SRWireless:SUMMary:WLAN:CCDF:ASCii?
:FETCh:SRWireless:PACKet:WLAN:CCDF:ASCii?

Command

```
:CONFigure:SRWireless:SELect:WLAN:CCDF
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:CCDF?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:WLAN:CCDF ON
```

:CONFigure:SRWireless:SELect:WLAN:CESTimation

Function

Enable/disable the availability of channel estimation values for OFDM signals.

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:WLAN:OFDM:CESTimation?
:FETCh:SRWireless:SUMMary:WLAN:OFDM:CESTimation:AMPLitude?
:FETCh:SRWireless:SUMMary:WLAN:OFDM:CESTimation:PHASe?
:FETCh:SRWireless:PACKet:WLAN:OFDM:CESTimation?
:FETCh:SRWireless:PACKet:WLAN:OFDM:CESTimation:AMPLitude?
:FETCh:SRWireless:PACKet:WLAN:OFDM:CESTimation:PHASe?

Command

```
:CONFigure:SRWireless:SELect:WLAN:CESTimation
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:CESTimation?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:WLAN:CEST ON
```

5 Remote Operation

:CONFigure:SRWireless:SELect:WLAN:CONSt

Function

Enable / disable WLAN constellation (graphical) measurements.

For details of measurements in this group see:
:FETCh:SRWireless:PACKet:WLAN:OFDM:CONSt[:BINary]?
:FETCh:SRWireless:PACKet:WLAN:DSSS:CONSt[:BINary]?

Command

```
:CONFigure:SRWireless:SELect:WLAN:CONSt
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:CONSt?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:WLAN:CONS ON
```

:CONFigure:SRWireless:SELect:WLAN:CPOWer

Function

Enable / disable sequential and true MIMO cross power measurements.

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:WLAN:CPOWer?
:FETCh:SRWireless:PACKet:WLAN:CPOWer?

Command

```
:CONFigure:SRWireless:SELect:WLAN:CPOWer
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:CPOWer?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:WLAN:CPOW ON
```

:CONFigure:SRWireless:SELect:WLAN:DSSS:PRAMps

Function

Enable / disable power on and power down ramp measurements – DSSS packets.

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:WLAN:DSSS:PRAMps?
:FETCh:SRWireless:PACKet:WLAN:DSSS:PRAMps?

Command

```
:CONFigure:SRWireless:SELect:WLAN:DSSS:PRAMps
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:DSSS:PRAMps?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:WLAN:DSSS:PRAM ON
```

:CONFigure:SRWireless:SELect:WLAN:EVM[:BASic]

Function

Enable / disable WLAN EVM measurements.

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM?
:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM?
:FETCh:SRWireless:SUMMary:WLAN:OFDM:AEVM?
:FETCh:SRWireless:PACKet:WLAN:OFDM:AEVM?
:FETCh:SRWireless:SUMMary:WLAN:OFDM:EVM?
:FETCh:SRWireless:PACKet:WLAN:OFDM:EVM?

Command

```
:CONFigure:SRWireless:SELect:WLAN:EVM[:BASic]
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:EVM[:BASic]?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	See note

Note:
The default setting depends on the segment type:

Any “AUTO” or “MIMO” segment types	ON
Any fixed standard segment types (e.g. WLG, WLAC)	OFF

Example of Use

```
:CONF:SRW:SEL:WLAN:EVM ON
```

5

Remote Operation

:CONFigure:SRWireless:SELect:WLAN:EVM:COMBined

Function

Enable / disable WLAN combined EVM measurements.
Combined EVM measurements are available for 802.11ac 80+80 signals, where a combined EVM of the upper and lower frequency segments is calculated.

For details of measurements in this group see:
`:FETCh:SRWireless:SUMMary:WLAN:OFDM:EVM:COMBined?`
`:FETCh:SRWireless:PACKet:WLAN:OFDM:EVM:COMBined?`

Command

`:CONFigure:SRWireless:SELect:WLAN:EVM:COMBined`

Query

`:CONFigure:SRWireless:SELect:WLAN:EVM:COMBined?`

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

`:CONF:SRW:SEL:WLAN:EVM:COMBined ON`

:CONFigure:SRWireless:SELect:WLAN:MASK:FULL

Function

Enable or disable full span spectral mask measurements. This function affects the span over which spectrum measurements are checked against the mask limits. By default this setting is enabled and the mask is applied across the full span of the measurement, in compliance with the IEEE standard. The spans over which spectral mask measurements are made are shown in the following table:

Standard	Full mask enabled - default (Full span used for mask tests)	Full mask disabled (Limited span used for mask tests)
802.11b	±35 MHz	±22 MHz
802.11a,g,n,ac,ax (20 MHz)	±35 MHz	±30 MHz
802.11n,ac,ax (40 MHz)	±65 MHz	±60 MHz
802.11ac,ax (80 MHz)	±140 MHz	±120 MHz
802.11ac (160 MHz)	±240 MHz	±240 MHz

Note:

The 802.11ac (80 MHz, 160MHz) and 802.11ax (80 MHz) spans assume that full spectrum measurements have been enabled.
(See `:CONFigure:SRWireless:SELect:WLAN:SPECtrum:FULL`.) If full span spectrum measurements are not enabled, testing against the mask limits will be limited to a span of ±80 MHz.

Command

```
:CONFigure:SRWireless:SELect:WLAN:MASK:FULL
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:MASK:FULL?
```

Response

#	Description	Format
1	Full span spectral mask measurements enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Full span spectral mask measurements enabled / disabled	Boolean	---	---	---	ON

Example of Use

```
:CONF:SRW:SEL:WLAN:MASK:FULL OFF          // Restrict the span over which
                                          // the mask is applied.
```


Remote Operation

5

:CONFigure:SRWireless:SELect:WLAN:OFDM:FERRor

Function

Enable / disable OFDM preamble frequency error v time measurements.

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:WLAN:OFDM:FERRor?
:FETCh:SRWireless:PACKet:WLAN:OFDM:FERRor?

Command

```
:CONFigure:SRWireless:SELect:WLAN:OFDM:FERRor
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:OFDM:FERRor?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:WLAN:OFDM:FERR ON
```

:CONFigure:SRWireless:SELect:WLAN:OFDM:PERRor

Function

Enable / disable OFDM phase error v symbol measurements.

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:WLAN:OFDM:PERRor?
:FETCh:SRWireless:PACKet:WLAN:OFDM:PERRor?

Command

```
:CONFigure:SRWireless:SELect:WLAN:OFDM:PERRor
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:OFDM:PERRor?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:WLAN:OFDM:PERR ON
```

:CONFigure:SRWireless:SELect:WLAN:POWer

Function

Enable / disable WLAN transmit power measurements.

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:WLAN:POWer?
:FETCh:SRWireless:PACKet:WLAN:POWer?

Command

```
:CONFigure:SRWireless:SELect:WLAN:POWer
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:POWer?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	See note

Note:
The default setting depends on the segment type:

Any “AUTO” or “MIMO” segment types	ON
Any fixed standard segment types (e.g. WLG, WLAC)	OFF

Example of Use

```
:CONFSRW:SEL:WLAN:POW ON
```

:CONFigure:SRWireless:SELect:WLAN:PPRofile

Function

Enable / disable WLAN power profile (graphical) measurements.

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:WLAN:PPRofile:AVERage[:BINary]?
:FETCh:SRWireless:SUMMary:WLAN:PPRofile:AVERage:ASCII?
:FETCh:SRWireless:PACKet:WLAN:PPRofile:AVERage[:BINary]?
:FETCh:SRWireless:PACKet:WLAN:PPRofile:AVERage:ASCII?
:FETCh:SRWireless:SUMMary:WLAN:PPRofile:PEAK[:BINary]?
:FETCh:SRWireless:SUMMary:WLAN:PPRofile:PEAK:ASCII?
:FETCh:SRWireless:PACKet:WLAN:PPRofile:PEAK[:BINary]?
:FETCh:SRWireless:PACKet:WLAN:PPRofile:PEAK:ASCII?

Command

```
:CONFigure:SRWireless:SELect:WLAN:PPRofile
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:PPRofile?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:WLAN:PPR ON
```

5

Remote Operation

:CONFigure:SRWireless:SELect:WLAN:PRAMps

Function

Enable / disable power on and power down ramp measurements on WLAN packets (both DSSS and OFDM).

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:WLAN:PRAMps?
:FETCh:SRWireless:PACKet:WLAN:PRAMps?

Note:
This command supersedes `:CONFigure:SRWireless:SELect:WLAN:DSSS:PRAMps` following a change to support power ramp measurements on OFDM as well as DSSS signals. The use of the older command is now deprecated.

Command

```
:CONFigure:SRWireless:SELect:WLAN:PRAMps
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:PRAMps?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:WLAN:PRAM ON
```

:CONFigure:SRWireless:SELect:WLAN:SPECtrum:FULL

Function

Enable full span 802.11ac (80 MHz, 160 MHz) measurements and 802.11ax (80 MHz) measurements. When enabled, spectrum measurements returned by the instrument will cover a span of ±140 MHz for 802.11ac, ax (80 MHz) and ±240 MHz for 802.11ac (160 MHz). When this setting is disabled, spectrum measurements of 802.11ac, ax signals will be limited to a maximum span of ±80 MHz. The setting is disabled by default.

Note:

Full span spectrum measurements are obtained by tuning the instrument's receiver to frequencies above and below the carrier and acquiring measurement data separately at each frequency. For this reason, measurement acquisition time will be longer when this function is enabled.

In the case of 802.11ac 80+80 signals, full span spectrum measurements are always enabled, regardless of this setting.

Command

```
:CONFigure:SRWireless:SELect:WLAN:SPECtrum:FULL
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:SPECtrum:FULL?
```

Response

#	Description	Format	Units
1	Full span 802.11ac (80 MHz, 160 MHz), 802.11ax (80 MHz) measurements enabled / disabled	Boolean	---

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Full span 802.11ac (80 MHz, 160 MHz), 802.11ax (80 MHz) measurements enabled / disabled	Boolean	---	---	---	OFF	----

Example of Use

```
:CONF:SRW:SEL:WLAN:SPEC:FULL ON
```

5 Remote Operation

:CONFigure:SRWireless:SELect:WLAN:SPECtrum:GRAPh

Function

Enable / disable WLAN transmitter spectrum (graphical) measurements.

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:GRAPh[:BINary]?
:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:GRAPh:ASCII?
:FETCh:SRWireless:PACKet:WLAN:SPECtrum:GRAPh[:BINary]?
:FETCh:SRWireless:PACKet:WLAN:SPECtrum:GRAPh:ASCII?

Command

```
:CONFigure:SRWireless:SELect:WLAN:SPECtrum:GRAPh
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:SPECtrum:GRAPh?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:SEL:WLAN:SPEC:GRAP ON
```

:CONFigure:SRWireless:SELect:WLAN:SPECtrum:NUMeric

Function

Enable / disable WLAN transmitter spectrum numeric measurements.

For details of measurements in this group see:
:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:NUMeric?
:FETCh:SRWireless:PACKet:WLAN:SPECtrum:NUMeric?

Command

```
:CONFigure:SRWireless:SELect:WLAN:SPECtrum:NUMeric
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:SPECtrum:NUMeric?
```

Response

#	Description	Format
1	Measurement enabled / disabled	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Measurement enabled / disabled.	Boolean	---	---	---	See note

Note:
The default setting depends on the segment type:

Any “AUTO” or “MIMO” segment types	ON
Any fixed standard segment types (e.g. WLG, WLAC)	OFF

Example of Use

```
:CONF:SRW:SEL:WLAN:SPEC:NUM ON
```

5 Remote Operation

:CONFigure:SRWireless:SELect:WLAN:SPOWer

Function

Enable / disable composite MIMO spatial stream power measurements.

Note:

This setting only applies to composite MIMO measurements and is enabled by default if the segment type is COMPMIMO. The setting has no effect on other measurements and returns OFF by default.

Command

```
:CONFigure:SRWireless:SELect:WLAN:SPOWer
```

Query

```
:CONFigure:SRWireless:SELect:WLAN:SPOWer?
```

Response

#	Description	Format	Units
1	Measurement enabled / disabled	Boolean	---

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Measurement enabled / disabled	Boolean	---	---	---	ON	----

Example of Use

```
:CONF:SRW:SEL:WLAN:SPOW OFF
```

:CONFigure:SRWireless:SPECtrum:WINDow

Function

Set the window type used for spectrum analysis measurements in the active segment.

The two window types are flat top and Gaussian.

The flat top window has the effect of merging adjacent frequency bins together to produce a spectrum with a smoother appearance. There is a loss of absolute power measurement accuracy, but the relative power level of closely-spaced signals is faithfully reproduced.
The Gaussian window gives the most accurate absolute power measurement at a spot frequency provided that the signal being measured coincides with the centre of a frequency bin. However if the signal lies midway between bins, the power measurement accuracy is lower than would be obtained with a flat top window because of the sharp cut-off of this type of filter. For the same reason, the relative power level of closely-spaced signals is less accurately reproduced, but there is less leakage between adjacent bins so that finer detail can be seen.

Command

```
:CONFigure:SRWireless:SPECtrum:WINDow
```

Query

```
:CONFigure:SRWireless:SPECtrum:WINDow?
```

Response

#	Description	Format	Units	Resolution
1	Window type	See definition	---	---

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Window type	See definition	---	---	---	FLATTOP	---

Window type:

FLATTOP	Flat top window.
GAUSSIAN	Gaussian window.

Example of Use

```
:CONF:SRW:SPEC:WIND GAUSSIAN        // Select the Gaussian window.
```

:CONFigure:SRWireless:STANdard?

Function

Returns the wireless standard and optional ID string defined for the active segment.

Command

N/A

Query

```
:CONFigure:SRWireless:STANdard?
```

Response

#	Description	Format
1	Wireless standard defined for the active segment.	See definition
2	User-defined ID string for the active segment.	String (up to 1024 bytes)

Wireless standard:

CW	CW
WLA	WLAN 802.11a
WLB	WLAN 802.11b
WLG	WLAN 802.11g
WLP	WLAN 802.11p
WLN	WLAN 802.11n
WLAC	WLAN 802.11ac
WLAX	WLAN 802.11ax
BT	Bluetooth
BLE	Bluetooth Low Energy
AUTOOFDM	Auto-ID OFDM packets
AUTODSSS	Auto-ID DSSS packets
AUTOBT	Auto-ID Bluetooth BR, EDR and LE packets
COMPMIMO	A composite MIMO measurement segment.
SEQMIMO	A sequential MIMO measurement segment.
TMIMOM	A true MIMO “M” segment.
TMIMOS	A true MIMO “S” segment.
SEQ80_80SISO	WLAN 802.11ac sequential 80+80 SISO
CON80_80SISOM	WLAN 802.11ac concurrent 80+80 SISO (“M” module control)
CON80_80SISOS	WLAN 802.11ac concurrent 80+80 SISO (“S” module control)
CON80_80TMIMOM	WLAN 802.11ac concurrent 80+80 True MIMO (“M” module control)
CON80_80TMIMOS	WLAN 802.11ac concurrent 80+80 True MIMO (“S” module control)
SEQ80_80TMIMOM	WLAN 802.11ac sequential 80+80 True MIMO (“M” module control)
SEQ80_80TMIMOS	WLAN 802.11ac sequential 80+80 True MIMO (“S” module control)
NONE	No segments have been defined.

Parameters

N/A

Example of Use

```
:CONF:SRW:STAN?
> WLG, 802.11g Segment
```

:CONFigure:SRWireless:STIMe

Function

Set or query the settling time for the active segment. (Time is from start of segment). Data captured during the settling time is ignored and does not contribute to measurements.

Command

```
:CONFigure:SRWireless:STIMe
```

Query

```
:CONFigure:SRWireless:STIMe?
```

Response

#	Description	Format	Units
1	Settling time	Real number	s

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Settling time	Number	s	0	1 s	0 s	1 µs

Example of Use

```
:CONF:SRW:STIM 500E-06      // Ignore the first 500 µs of the segment capture
                            // data to allow for settling.
```

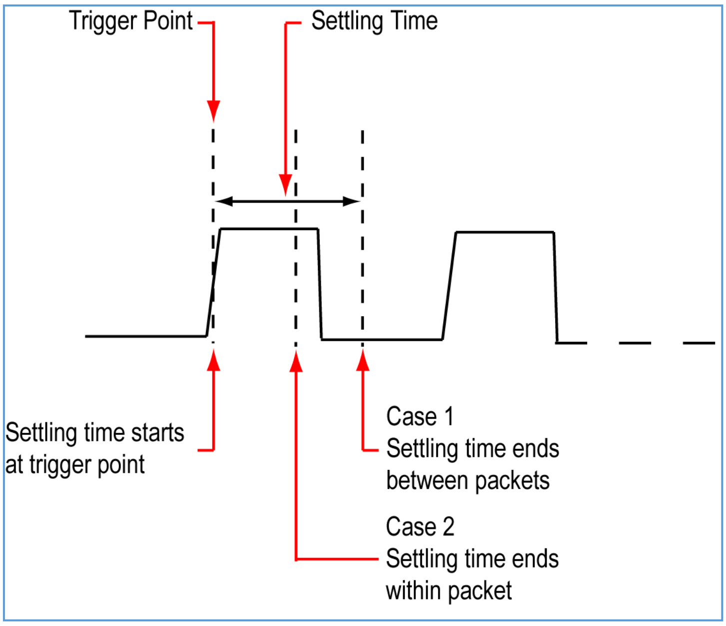
:CONFigure:SRWireless:STPSync

Function

Enable / disable the settling time packet synchronisation function.
This function is designed to ensure that a fragment of a packet captured at the start of a segment is skipped and that analysis begins at the start of the next complete packet.

This command is useful under the following conditions:

- Capture mode is TIME (see :CONF:SRW:CAPT:MODE)
- Trigger mode is LEVEL (see :CONF:SRW:TRIG)
- Settling time is greater than zero (see :CONF:SRW:STIM)



In level trigger mode under default conditions, the trigger point occurs on the rising edge of a packet. The system is designed so that under these conditions the packet that caused the trigger can be measured successfully. However if a settling time is used, analysis of the captured data may start part way through a packet (see diagram). This will result in an error being reported.
This error condition can be avoided by ensuring that the settling time is long enough to skip the first packet and end during the gap between packets. However the settling time can be difficult to set up if the packet duration and inter-packet gap are not known or prone to jitter. :CONFigure:SRWireless:STPSync is designed to overcome this problem. When this function is enabled, the system automatically searches for the first packet rising edge after the settling time has elapsed. This ensure that if the settling time ends within a packet, that packet is not analysed.

This command is designed to work when the capture mode is set to TIME. If it is set to PACKET it is possible to use CONFigure:SRW:PACKets:SKIP to set up the capture hardware to skip a whole number of packets from the start of the capture. For that reason packet count mode is usually more convenient to use than timed captures.

Command

```
:CONFigure:SRWireless:STPSync
```

Query

```
:CONFigure:SRWireless:STPSync?
```

Response

#	Description	Format
1	Settling time packet synchronisation	Boolean

Parameters

#	Description	Format	Units	Min	Max	Default
1	Settling time packet synchronisation	Boolean	---	---	---	OFF

Example of Use

```
:CONF:SRW:STPS ON             // Enable settling time packet synchronisation.
```

:CONFigure:SRWireless:TDELay

Function

Set or query the trigger delay time for the active segment. This is the time between the trigger being detected and the start of data acquisition. When the delay is negative, capture starts before the trigger; when positive it starts after the trigger.

Command

```
:CONFigure:SRWireless:TDELay
```

Query

```
:CONFigure:SRWireless:TDELay?
```

Response

#	Description	Format	Units
1	Trigger delay	Real number	s

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Trigger delay	Number+	s	–1.33	1.33	0	1 μs

Example of Use

```
:CONF:SRW:TDEL -5us          // Set the trigger delay to allow data
                             // capture 5 μs before the trigger event.
```

:CONFigure:SRWireless:TIME

Function

Set or query the duration of the active segment.

Command

```
:CONFigure:SRWireless:TIME
```

Query

```
:CONFigure:SRWireless:TIME?
```

Response

#	Description	Format	Units
1	Duration of the active segment	Real number	s

Parameters

#	Description	Format	Units	Min	Max	Default
1	Duration of the active segment	Number+	s	50 µs	See notes	10 ms

Note:

Max depends on total time remaining for capture. :CONF:SRW:CAPT:TIME? can be used to determine this.

Example of Use

```
:CONF:SRW:TIME 10ms
```

:CONFigure:SRWireless:TLEVel

Function

Query the trigger level (relative to the expected power level) for the active segment.

Command

```
:CONFigure:SRWireless:TLEVel
```

Query

```
:CONFigure:SRWireless:TLEVel?
```

Response

#	Description	Format	Units
1	Trigger level relative to expected power level.	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Trigger level relative to expected power level.	Number+	dB	–60	0	–20	1 dB

Example of Use

```
:CONF:SRW:TLEV -25          // Set the trigger level 25 dB below the
                            // expected power level.
```

:CONFigure:SRWireless:TMIMo:CONNect

Function

Connects to all T-MIMO “S” modules using the IP addresses assigned in the TMIMOS segments and transfers configuration information.

Note:
A TMIMOM segment and at least one TMIMOS segment must have been created before this command is used.

Command

```
:CONFigure:SRWireless:TMIMo:CONNect
```

Query

N/A

Response

N/A

Parameters

N/A

Example of Use

```
:CONF:SRW:TMIM:CONN
```

:CONFigure:SRWireless:TMIMo:DISConnect

Function

Cancels the connections to all T-MIMO “S” modules.

Notes:

1. A TMIMOM segment and at least one TMIMOS segment must have been created before this command is used.
2. The command has no effect if there are no connections established.

Command

```
:CONFigure:SRWireless:TMIMo:DISConnect
```

Query

N/A

Response

N/A

Parameters

N/A

Example of Use

```
:CONF:SRW:TMIM:DISC
```

:CONFigure:SRWireless:TRIGger

Function

Set or query the trigger mode for capturing the active segment.

Command

```
:CONFigure:SRWireless:TRIGger
```

Query

```
:CONFigure:SRWireless:TRIGger?
```

Response

#	Description	Format
1	Trigger mode	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	Trigger mode	See definition	---	---	---	IMMEDIATE

Trigger mode:

IMMEDIATE	Capture starts regardless of input level.
LEVEL	Capture starts when RF exceeds a threshold level.

Example of Use

```
:CONF:SRW:TRIG LEVEL        // Enable triggering based on RF level
```

:CONFigure:SRWireless:TTIMeout

Function

Set the trigger timeout for use when the trigger mode is set to Level.

If the timeout is set to zero seconds, the system will wait indefinitely for a trigger. If it is set to a number of seconds greater than zero, the capture will be aborted if a trigger is not detected within the timeout period.

The timeout count starts on receipt of the INIT:SRW command. If a timeout occurs:

1. The capture is aborted.
2. Bit 2 of the response returned by :STAT:SRW:MEAS? is set.
3. Capture information returned by :FETC:SRW:CINF includes the Trigger Timeout error code - 517 (0x205).
4. LED 3 on the front panel of the MU887000A module changes to red, and remains red until the next capture is initiated.

Command

```
:CONFigure:SRWireless:TTIMeout
```

Query

```
:CONFigure:SRWireless:TTIMeout?
```

Response

#	Description	Format	Units	Resolution
1	Trigger timeout	Integer	s	1 s

Parameters

#	Description	Format	Units	Min	Max	Def	Resolution
1	Trigger timeout	Integer	s	0	60	0	1 s

Example of Use

```
:CONF:SRW:TTIM 10     // Set the trigger timeout to 10s
:INIT:SRW             // Initiate a capture.
*WAI
:FETC:SRW:CINF?       // Check capture information to determine whether
                      // a trigger timeout occurred.
> 517,…               // First field (capture status) returns trigger
                      // timeout error 517 (0x205).
```

5 Remote Operation

:CONFigure:SRWireless:WLA:DRATe

Function

Set or query the expected data rate of WLAN A packets in this segment.

Command

```
:CONFigure:SRWireless:WLA:DRATe
```

Query

```
:CONFigure:SRWireless:WLA:DRATe?
```

Response

#	Description	Format
	WLAN A data rate	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN A data rate	See definition	---	---	---	54MBPS

WLAN A data rate:

6MBPS	6 Mbps
9MBPS	9 Mbps
12MBPS	12 Mbps
18MBPS	18 Mbps
24MBPS	24 Mbps
36MBPS	36 Mbps
48MBPS	48 Mbps
54MBPS	54 Mbps

Example of Use

```
:CONF:SRW:WLA:DRAT 6MBPS
```

:CONFigure:SRWireless:WLA:EVM:ULIMit

Function

WLAN A EVM upper limits.

Command

```
:CONFigure:SRWireless:WLA:EVM:ULIMit
```

Note:

If any parameter is out of range, the command will fail and all the original parameter values will be retained.

Query

```
:CONFigure:SRWireless:WLA:EVM:ULIMit?
```

Response

#	Description	Format	Units
1	WLAN A EVM upper limit (6 Mbps)	Real number	dB
2	WLAN A EVM upper limit (9 Mbps)	Real number	dB
3	WLAN A EVM upper limit (12 Mbps)	Real number	dB
4	WLAN A EVM upper limit (18 Mbps)	Real number	dB
5	WLAN A EVM upper limit (24 Mbps)	Real number	dB
6	WLAN A EVM upper limit (36 Mbps)	Real number	dB
7	WLAN A EVM upper limit (48 Mbps)	Real number	dB
8	WLAN A EVM upper limit (54 Mbps)	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	WLAN A EVM upper limit (6 Mbps)	Number+	dB	–100	100	–5	0.1 dB
2	WLAN A EVM upper limit (9 Mbps)	Number+	dB	–100	100	–8	0.1 dB
3	WLAN A EVM upper limit (12 Mbps)	Number+	dB	–100	100	–10	0.1 dB
4	WLAN A EVM upper limit (18 Mbps)	Number+	dB	–100	100	–13	0.1 dB
5	WLAN A EVM upper limit (24 Mbps)	Number+	dB	–100	100	–16	0.1 dB
6	WLAN A EVM upper limit (36 Mbps)	Number+	dB	–100	100	–19	0.1 dB
7	WLAN A EVM upper limit (48 Mbps)	Number+	dB	–100	100	–22	0.1 dB
8	WLAN A EVM upper limit (54 Mbps)	Number+	dB	–100	100	–25	0.1 dB

Example of Use

```
:CONF:SRW:WLA:EVM:ULIM -5,-8,-10,-13,-16,-19,-22,-25
```

:CONFigure:SRWireless:WLA:SCFTolerance:LIMit

Function

Set or query the WLAN A (OFDM) symbol clock frequency tolerance limit (±).

Command

```
:CONFigure:SRWireless:WLA:SCFTolerance:LIMit
```

Query

```
:CONFigure:SRWireless:WLA:SCFTolerance:LIMit?
```

Response

#	Description	Format	Units
	Symbol clock frequency tolerance limit (±)	Real number	ppm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Symbol clock frequency tolerance limit (±)	Number+	ppm	0	200	20	0.1 ppm

Example of Use

```
:CONF:SRW:WLA:SCFT:LIM 20          // Set to ±20 ppm.
```

:CONFigure:SRWireless:WLA:SFLatness:LIMit

Function

Set or query the WLAN A (OFDM) spectral flatness limits.

Command

```
:CONFigure:SRWireless:WLA:SFLatness:LIMit
```

Note:

If any parameter is out of range, the command will fail and all the original parameter values will be retained.

Query

```
:CONFigure:SRWireless:WLA:SFLatness:LIMit?
```

Response

#	Description	Format	Units
1	Upper limit: subcarriers –26 to –17	Real number	dB
2	Lower limit: subcarriers –26 to –17	Real number	dB
3	Upper limit: subcarriers –16 to –1	Real number	dB
4	Lower limit: subcarriers –16 to –1	Real number	dB
5	Upper limit: subcarriers +1 to +16	Real number	dB
6	Lower limit: subcarriers +1 to +16	Real number	dB
7	Upper limit: subcarriers +17 to +26	Real number	dB
8	Lower limit: subcarriers +17 to +26	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Upper limit: subcarriers –26 to –17	Number+	dB	–100	100	2	0.01 dB
2	Lower limit: subcarriers –26 to –17	Number+	dB	–100	100	–4	0.01 dB
3	Upper limit: subcarriers –6 to –1	Number+	dB	–100	100	2	0.01 dB
4	Lower limit: subcarriers –16 to –1	Number+	dB	–100	100	–2	0.01 dB
5	Upper limit: subcarriers +1 to +16	Number+	dB	–100	100	2	0.01 dB
6	Lower limit: subcarriers +1 to +16	Number+	dB	–100	100	–2	0.01 dB
7	Upper limit: subcarriers +17 to +26	Number+	dB	–100	100	2	0.01 dB
8	Lower limit: subcarriers +17 to +26	Number+	dB	–100	100	–4	0.01 dB

Example of Use

```
:CONF:SRW:WLA:SFL:LIM 2,-5,2,-2,2,-2,2,-5
```

:CONFigure:SRWireless:WLA:SMASk:ULIMit

Function

Define the spectral mask limits for WLAN IEEE 802.11a signals.

Command

```
:CONFigure:SRWireless:WLA:SMASk:ULIMit
```

Query

```
:CONFigure:SRWireless:WLA:SMASk:ULIMit?
```

Response

#	Description	Format	Units	Resolution
1	fc – 30 MHz mask break point	Real number	dBr	0.01 dB
2	fc – 20 MHz mask break point	Real number	dBr	0.01 dB
3	fc – 11 MHz mask break point	Real number	dBr	0.01 dB
4	fc + 11 MHz mask break point	Real number	dBr	0.01 dB
5	fc + 20 MHz mask break point	Real number	dBr	0.01 dB
6	fc + 30 MHz mask break point	Real number	dBr	0.01 dB

Parameters

#	Description	Format	Units	Min	Max	Def	Resolution
1	fc – 30 MHz mask break point	Number+	dBr	–100	0	–40	0.01 dB
2	fc – 20 MHz mask break point	Number+	dBr	–100	0	–28	0.01 dB
3	fc – 11 MHz mask break point	Number+	dBr	–100	0	–20	0.01 dB
4	fc + 11 MHz mask break point	Number+	dBr	–100	0	–20	0.01 dB
5	fc + 20 MHz mask break point	Number+	dBr	–100	0	–28	0.01 dB
6	fc + 30 MHz mask break point	Number+	dBr	–100	0	–40	0.01 dB

Example of Use

```
:CONF:SRW:WLA:SMAS:ULIM -40.00,-28.00,-20.00,-20.00,-28.00,-40.00
:CONF:SRW:WLA:SMAS:ULIM
> -40.00,-28.00,-20.00,-20.00,-28.00,-40.00
```

:CONFigure:SRWireless:WLA:TCFLeakage:ULIMit

Function

Set or query the WLAN A (OFDM) transmitter center frequency leakage upper limit.

Command

```
:CONFigure:SRWireless:WLA:TCFLeakage:ULIMit
```

Query

```
:CONFigure:SRWireless:WLA:TCFLeakage:ULIMit?
```

Response

#	Description	Format	Units
1	Transmitter center frequency leakage upper limit	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Transmitter center frequency leakage upper limit	Number+	dB	–100	100	–15	0.01 dB

Example of Use

```
:CONF:SRW:WLA:TCFL:ULIM  -14
```

:CONFigure:SRWireless:WLA:TCFTolerance:LIMit

Function

Set or query the WLAN A (OFDM) transmitter center frequency tolerance limit (±).

Command

```
:CONFigure:SRWireless:WLA:TCFTolerance:LIMit
```

Query

```
:CONFigure:SRWireless:WLA:TCFTolerance:LIMit?
```

Response

#	Description	Format	Units
1	Transmitter center frequency tolerance limit (±)	Real number	ppm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Transmitter center frequency tolerance limit (±)	Number+	ppm	0	200	20	0.1 ppm

Example of Use

```
:CONF:SRW:WLA:TCFT:LIM 19          // Set to ±19 ppm.
```

:CONFigure:SRWireless:WLAC:EVM:ULIMit

Function

Set or query the WLAN AC EVM upper limits

Command

```
:CONFigure:SRWireless:WLAC:EVM:ULIMit
```

Note:

If any parameter is out of range, the command will fail and all the original parameter values will be retained.

Query

```
:CONFigure:SRWireless:WLAC:EVM:ULIMit?
```

Response

#	Description	Format	Units
1	WLAN AC EVM upper limit (MCS 0)	Real number	dB
2	WLAN AC EVM upper limit (MCS 1)	Real number	dB
3	WLAN AC EVM upper limit (MCS 2)	Real number	dB
4	WLAN AC EVM upper limit (MCS 3)	Real number	dB
5	WLAN AC EVM upper limit (MCS 4)	Real number	dB
6	WLAN AC EVM upper limit (MCS 5)	Real number	dB
7	WLAN AC EVM upper limit (MCS 6)	Real number	dB
8	WLAN AC EVM upper limit (MCS 7)	Real number	dB
9	WLAN AC EVM upper limit (MCS 8)	Real number	dB
10	WLAN AC EVM upper limit (MCS 9)	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	WLAN AC EVM upper limit (MCS0)	Number+	dB	–100	100	–5	0.1 dB
2	WLAN AC EVM upper limit (MCS1)	Number+	dB	–100	100	–10	0.1 dB
3	WLAN AC EVM upper limit (MCS2)	Number+	dB	–100	100	–13	0.1 dB
4	WLAN AC EVM upper limit (MCS3)	Number+	dB	–100	100	–16	0.1 dB
5	WLAN AC EVM upper limit (MCS4)	Number+	dB	–100	100	–19	0.1 dB
6	WLAN AC EVM upper limit (MCS5)	Number+	dB	–100	100	–22	0.1 dB
7	WLAN AC EVM upper limit (MCS6)	Number+	dB	–100	100	–25	0.1 dB
8	WLAN AC EVM upper limit (MCS7)	Number+	dB	–100	100	–27	0.1 dB
9	WLAN AC EVM upper limit (MCS8)	Number+	dB	–100	100	–30	0.1 dB
10	WLAN AC EVM upper limit (MCS9)	Number+	dB	–100	100	–32	0.1 dB

Example of Use

```
:CONF:SRW:WLAC:EVM:ULIM -5,-10,-13,-16,-19,-22,-25,-27,-30,-32
```

:CONFigure:SRWireless:WLAC:GINTerval

Function

Set or query the guard interval of the expected WLAN AC packets in this segment.

Command

```
:CONFigure:SRWireless:WLAC:GINTerval
```

Query

```
:CONFigure:SRWireless:WLAC:GINTerval?
```

Response

#	Description	Format
1	WLAN AC Guard Interval	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN AC Guard Interval	See definition	---	---	---	LONG

WLAN AC Guard Interval:

SHORT	Short guard interval (400 ns)
LONG	Long guard interval (800 ns)

Example of Use

```
:CONF:SRW:WLAC:GINT SHORT
```

:CONFigure:SRWireless:WLAC:MCS

Function

Set or query the expected MCS of the WLAN AC packets in this segment.

Command

```
:CONFigure:SRWireless:WLAC:MCS
```

Query

```
:CONFigure:SRWireless:WLAC:MCS?
```

Response

#	Description	Format
1	WLAN AC MCS	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN AC MCS	See definition	---	---	---	MCS8

WLAN AC MCS:

MCS0	MCS 0	(BPSK 1/2)
MCS1	MCS 1	(QPSK 1/2)
MCS2	MCS 2	(QPSK 3/4)
MCS3	MCS 3	(16-QAM 1/2)
MCS4	MCS 4	(16-QAM 3/4)
MCS5	MCS 5	(64-QAM 2/3)
MCS6	MCS 6	(64-QAM 3/4)
MCS7	MCS 7	(64-QAM 5/6)
MCS8	MCS 8	(256-QAM 3/4)
MCS9	MCS 9	(256-QAM 5/6)

Example of Use

```
:CONF:SRW:WLAC:MCS MCS9
```

:CONFigure:SRWireless:WLAC:PTYPe

Function

Set or query the expected PPDU type of the WLAN AC packets in this segment.

Command

```
:CONFigure:SRWireless:WLAC:PTYPe
```

Query

```
:CONFigure:SRWireless:WLAC:PTYPe?
```

Response

#	Description	Format
1	WLAN AC PPDU Type	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN AC PPDU Type	See definition	---	---	---	VHT80

WLAN AC PPDU Type:

VHT20	20 MHz
VHT40	40 MHz
VHT80	80 MHz
VHT160	160 MHz

Example of Use

```
:CONF:SRW:WLAC:PTYP VHT160
```

:CONFigure:SRWireless:WLAC:SCFTolerance:LIMit

Function

Set or query the WLAN AC (OFDM) symbol clock frequency tolerance limit (±).

Command

```
:CONFigure:SRWireless:WLAC:SCFTolerance:LIMit
```

Query

```
:CONFigure:SRWireless:WLAC:SCFTolerance:LIMit?
```

Response

#	Description	Format	Units
	Symbol clock frequency tolerance limit (±)	Real number	ppm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Symbol clock frequency tolerance limit (±)	Number+	ppm	0	200	20	0.1 ppm

Example of Use

```
:CONF:SRW:WLAC:SCFT:LIM 19          // Set to ±19 ppm.
```

:CONFigure:SRWireless:WLAC:SFLatness:LIMit

Function

Set or query the WLAN AC (OFDM) spectral flatness limits

Command

`:CONFigure:SRWireless:WLAC:SFLatness:LIMit`

Note:

If any parameter is out of range, the command will fail and all the original parameter values will be retained.

Query

`:CONFigure:SRWireless:WLAC:SFLatness:LIMit?`

Response

#	Description	Format	Units
	20 MHz:		
1	Upper limit: subcarriers –28 to –17	Real number	dB
2	Lower limit: subcarriers –28 to –17	Real number	dB
3	Upper limit: subcarriers –16 to –1	Real number	dB
4	Lower limit: subcarriers –16 to –1	Real number	dB
5	Upper limit: subcarriers +1 to +16	Real number	dB
6	Lower limit: subcarriers +1 to +16	Real number	dB
7	Upper limit: subcarriers +17 to +28	Real number	dB
8	Lower limit: subcarriers +17 to +28	Real number	dB
	40 MHz:		
9	Upper limit: subcarriers –58 to –43	Real number	dB
10	Lower limit: subcarriers –58 to –43	Real number	dB
11	Upper limit: subcarriers –42 to –2	Real number	dB
12	Lower limit: subcarriers –42 to –2	Real number	dB
13	Upper limit: subcarriers +2 to +42	Real number	dB
14	Lower limit: subcarriers +2 to +42	Real number	dB
15	Upper limit: subcarriers +43 to +58	Real number	dB
16	Lower limit: subcarriers +43 to +58	Real number	dB
	80 MHz:		
17	Upper limit: subcarriers –122 to –85	Real number	dB
18	Lower limit: subcarriers –122 to –85	Real number	dB
19	Upper limit: subcarriers –84 to –2	Real number	dB
20	Lower limit: subcarriers –84 to –2	Real number	dB
21	Upper limit: subcarriers +2 to +84	Real number	dB
22	Lower limit: subcarriers +2 to +84	Real number	dB
23	Upper limit: subcarriers +85 to +122	Real number	dB
24	Lower limit: subcarriers +85 to +122	Real number	dB

	160 MHz:		
25	Upper limit: subcarriers –250 to –173	Real number	dB
26	Lower limit: subcarriers –250 to –173	Real number	dB
27	Upper limit: subcarriers –172 to –130	Real number	dB
28	Lower limit: subcarriers –172 to –130	Real number	dB
29	Upper limit: subcarriers –126 to –44	Real number	dB
30	Lower limit: subcarriers –126 to –44	Real number	dB
31	Upper limit: subcarriers –43 to –6	Real number	dB
32	Lower limit: subcarriers –43 to –6	Real number	dB
33	Upper limit: subcarriers +6 to +43	Real number	dB
34	Lower limit: subcarriers +6 to +43	Real number	dB
35	Upper limit: subcarriers +44 to +126	Real number	dB
36	Lower limit: subcarriers +44 to +126	Real number	dB
37	Upper limit: subcarriers +130 to +172	Real number	dB
38	Lower limit: subcarriers +130 to +172	Real number	dB
39	Upper limit: subcarriers +173 to +250	Real number	dB
40	Lower limit: subcarriers +173 to +250	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Def.	Resolution
	20 MHz:						
1	Upper limit: subcarriers –28 to –17	Number+	dB	–100	100	4	0.01 dB
2	Lower limit: subcarriers –28 to –17	Number+	dB	–100	100	–6	0.01 dB
3	Upper limit: subcarriers –16 to –1	Number+	dB	–100	100	4	0.01 dB
4	Lower limit: subcarriers –16 to –1	Number+	dB	–100	100	–4	0.01 dB
5	Upper limit: subcarriers +1 to +16	Number+	dB	–100	100	4	0.01 dB
6	Lower limit: subcarriers +1 to +16	Number+	dB	–100	100	–4	0.01 dB
7	Upper limit: subcarriers +17 to +28	Number+	dB	–100	100	4	0.01 dB
8	Lower limit: subcarriers +17 to +28	Number+	dB	–100	100	–6	0.01 dB
	40 MHz:						
9	Upper limit: subcarriers –58 to –43	Number+	dB	–100	100	4	0.01 dB
10	Lower limit: subcarriers –58 to –43	Number+	dB	–100	100	–6	0.01 dB
11	Upper limit: subcarriers –42 to –2	Number+	dB	–100	100	4	0.01 dB
12	Lower limit: subcarriers –42 to –2	Number+	dB	–100	100	–4	0.01 dB
13	Upper limit: subcarriers +2 to +42	Number+	dB	–100	100	4	0.01 dB
14	Lower limit: subcarriers +2 to +42	Number+	dB	–100	100	–4	0.01 dB
15	Upper limit: subcarriers +43 to +58	Number+	dB	–100	100	4	0.01 dB
16	Lower limit: subcarriers +43 to +58	Number+	dB	–100	100	–6	0.01 dB
	80 MHz:						
17	Upper limit: subcarriers –122 to –85	Number+	dB	–100	100	4	0.01 dB
18	Lower limit: subcarriers –122 to –85	Number+	dB	–100	100	–6	0.01 dB
19	Upper limit: subcarriers –84 to –2	Number+	dB	–100	100	4	0.01 dB
20	Lower limit: subcarriers –84 to –2	Number+	dB	–100	100	–4	0.01 dB
21	Upper limit: subcarriers +2 to +84	Number+	dB	–100	100	4	0.01 dB

22	Lower limit: subcarriers +2 to +84	Number+	dB	–100	100	–4	0.01 dB
23	Upper limit: subcarriers +85 to +122	Number+	dB	–100	100	4	0.01 dB
24	Lower limit: subcarriers +85 to +122	Number+	dB	–100	100	–6	0.01 dB
	160 MHz:						
25	Upper limit: subcarriers –250 to –173	Number+	dB	–100	100	4	0.01 dB
26	Lower limit: subcarriers –250 to –173	Number+	dB	–100	100	–6	0.01 dB
27	Upper limit: subcarriers –172 to –130	Number+	dB	–100	100	4	0.01 dB
28	Lower limit: subcarriers –172 to –130	Number+	dB	–100	100	–4	0.01 dB
29	Upper limit: subcarriers –126 to –44	Number+	dB	–100	100	4	0.01 dB
30	Lower limit: subcarriers –126 to –44	Number+	dB	–100	100	–4	0.01 dB
31	Upper limit: subcarriers –43 to –6	Number+	dB	–100	100	4	0.01 dB
32	Lower limit: subcarriers –43 to –6	Number+	dB	–100	100	–6	0.01 dB
33	Upper limit: subcarriers +6 to +43	Number+	dB	–100	100	4	0.01 dB
34	Lower limit: subcarriers +6 to +43	Number+	dB	–100	100	–6	0.01 dB
35	Upper limit: subcarriers +44 to +126	Number+	dB	–100	100	4	0.01 dB
36	Lower limit: subcarriers +44 to +126	Number+	dB	–100	100	–4	0.01 dB
37	Upper limit: subcarriers +130 to +172	Number+	dB	–100	100	4	0.01 dB
38	Lower limit: subcarriers +130 to +172	Number+	dB	–100	100	–4	0.01 dB
39	Upper limit: subcarriers +173 to +250	Number+	dB	–100	100	4	0.01 dB
40	Lower limit: subcarriers +173 to +250	Number+	dB	–100	100	–6	0.01 dB

Example of Use

```
:CONF:SRW:WLAC:SFL:LIM 4,-6,4,-4,4,-4,4,-6,
                       4,-6,4,-4,4,-4,4,-6,
                       4,-6,4,-4,4,-4,4,-6,
                       4,-6,4,-4,4,-4,4,-6,4,-6,4,-4,4,-4,4,-6
```

:CONFigure:SRWireless:WLAC:SMASk:ULIMit

Function

Define the spectral mask limits for WLAN IEEE 802.11ac signals.

Command

```
:CONFigure:SRWireless:WLAC:SMASk:ULIMit
```

Query

```
:CONFigure:SRWireless:WLAC:SMASk:ULIMit
```

Response

#	Description	Format	Units	Resolution
1	fc – 30 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
2	fc – 20 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
3	fc – 11 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
4	fc + 11 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
5	fc + 20 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
6	fc + 30 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
7	fc – 60 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
8	fc – 40 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
9	fc – 21 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
10	fc + 21 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
11	fc + 40 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
12	fc + 60 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
13	fc – 120 MHz mask break point (80 MHz)	Real number	dBr	0.01 dB
14	fc – 80 MHz mask break point (80 MHz)	Real number	dBr	0.01 dB
15	fc – 41 MHz mask break point (80 MHz)	Real number	dBr	0.01 dB
16	fc + 41 MHz mask break point (80 MHz)	Real number	dBr	0.01 dB
17	fc + 80 MHz mask break point (80 MHz)	Real number	dBr	0.01 dB
18	fc + 120 MHz mask break point (80 MHz)	Real number	dBr	0.01 dB
19	fc – 240 MHz mask break point (160 MHz)	Real number	dBr	0.01 dB
20	fc – 160 MHz mask break point (160 MHz)	Real number	dBr	0.01 dB
21	fc – 81 MHz mask break point (160 MHz)	Real number	dBr	0.01 dB
22	fc + 81 MHz mask break point (160 MHz)	Real number	dBr	0.01 dB
23	fc + 160 MHz mask break point (160 MHz)	Real number	dBr	0.01 dB
24	fc + 240 MHz mask break point (160 MHz)	Real number	dBr	0.01 dB

5

Remote Operation

Parameters

#	Description	Format	Units	Min	Max	Def	Res
1	fc – 30 MHz mask break pt (20 MHz)	Number+	dBr	–100	0	–40	0.01 dB
2	fc – 20 MHz mask break pt (20 MHz)	Number+	dBr	–100	0	–28	0.01 dB
3	fc – 11 MHz mask break pt (20 MHz)	Number+	dBr	–100	0	–20	0.01 dB
4	fc + 11 MHz mask break pt (20 MHz)	Number+	dBr	–100	0	–20	0.01 dB
5	fc + 20 MHz mask break pt (20 MHz)	Number+	dBr	–100	0	–28	0.01 dB
6	fc + 30 MHz mask break pt (20 MHz)	Number+	dBr	–100	0	–40	0.01 dB
7	fc – 60 MHz mask break pt (40 MHz)	Number+	dBr	–100	0	–40	0.01 dB
8	fc – 40 MHz mask break pt (40 MHz)	Number+	dBr	–100	0	–28	0.01 dB
9	fc – 21 MHz mask break pt (40 MHz)	Number+	dBr	–100	0	–20	0.01 dB
10	fc + 21 MHz mask break pt (40 MHz)	Number+	dBr	–100	0	–20	0.01 dB
11	fc + 40 MHz mask break pt (40 MHz)	Number+	dBr	–100	0	–28	0.01 dB
12	fc + 60 MHz mask break pt (40 MHz)	Number+	dBr	–100	0	–40	0.01 dB
13	fc – 120 MHz mask break pt (80 MHz)	Number+	dBr	–100	0	–40	0.01 dB
14	fc – 80 MHz mask break pt (80 MHz)	Number+	dBr	–100	0	–28	0.01 dB
15	fc – 41 MHz mask break pt (80 MHz)	Number+	dBr	–100	0	–20	0.01 dB
16	fc + 41 MHz mask break pt (80 MHz)	Number+	dBr	–100	0	–20	0.01 dB
17	fc + 80 MHz mask break pt (80 MHz)	Number+	dBr	–100	0	–28	0.01 dB
18	fc + 120 MHz mask break pt (80 MHz)	Number+	dBr	–100	0	–40	0.01 dB
19	fc – 240 MHz mask break pt (160 MHz)	Number+	dBr	–100	0	–40	0.01 dB
20	fc – 160 MHz mask break pt (160 MHz)	Number+	dBr	–100	0	–28	0.01 dB
21	fc – 81 MHz mask break pt (160 MHz)	Number+	dBr	–100	0	–20	0.01 dB
22	fc + 81 MHz mask break pt (160 MHz)	Number+	dBr	–100	0	–20	0.01 dB
23	fc + 160 MHz mask break pt (160 MHz)	Number+	dBr	–100	0	–28	0.01 dB
24	fc + 240 MHz mask break pt (160 MHz)	Number+	dBr	–100	0	–40	0.01 dB

Example of Use

```
:CONF:SRW:WLAC:SMAS:ULIM -40.00,-28.00,-20.00,-20.00,-28.00,-40.00,
                         -40.00,-28.00,-20.00,-20.00,-28.00,-40.00
                         -40.00,-28.00,-20.00,-20.00,-28.00,-40.00
                         -40.00,-28.00,-20.00,-20.00,-28.00,-40.00

:CONF:SRW:WLAC:SMAS:ULIM?
> -40.00,-28.00,-20.00,-20.00,-28.00,-40.00,
-40.00,-28.00,-20.00,-20.00,-28.00,-40.00
-40.00,-28.00,-20.00,-20.00,-28.00,-40.00
-40.00,-28.00,-20.00,-20.00,-28.00,-40.00
```

:CONFigure:SRWireless:WLAC:SPECtrum:C80_80

Function

Specify whether the IEEE 802.11ac 80+80 frequency segments are separate or combined.

For most DUTs the primary and secondary frequency segments of an 802.11ac transmission are combined in a single antenna port. (In the case of a MIMO device, there will more than one antenna port, but each port will carry both frequency segments of the 80+80 transmission.) Some DUTs, however, transmit the two frequency segments on separate antenna ports. The SRW Application needs to know which method is used in the DUT so that it can calculate the combined spectrum correctly.

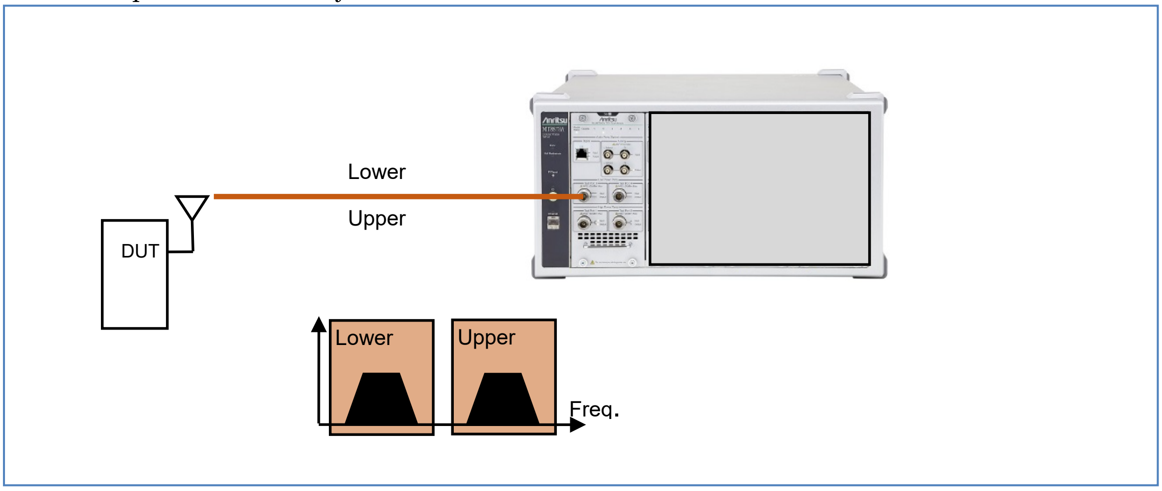


Figure 5.4-1 - Combined 80+80 Transmission

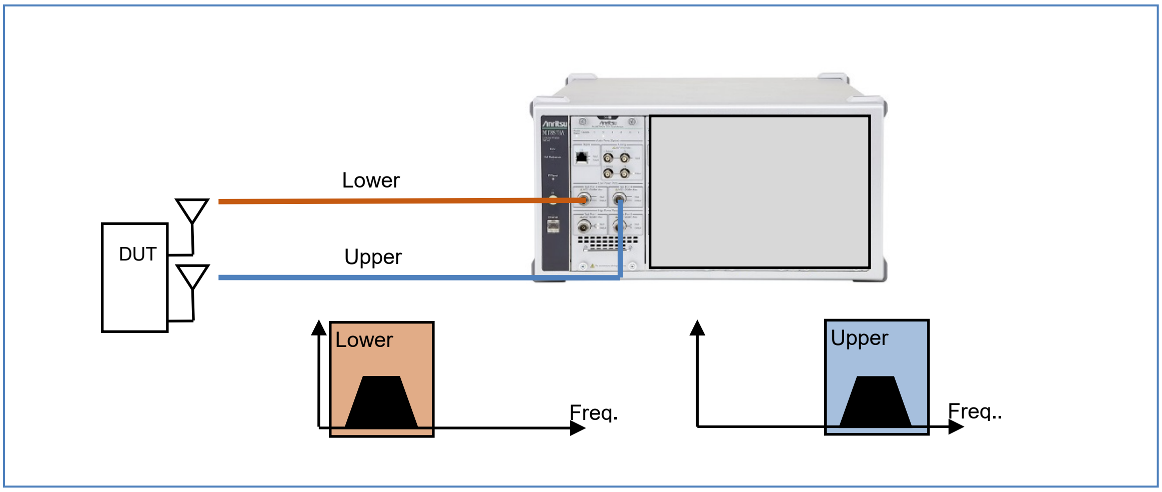


Figure 5.4-2 - Separate 80+80 Transmission

Set this setting to SEPARATE if the DUT outputs the two 80+80 frequency segments on separate antenna ports and these signals are taken to separate inputs on the MT8870A (Figure 5.4-2). Set it to COMBINED if the frequency segments are combined in the DUT

(Figure 5.4-1) or **externally using a combiner**. The default setting is COMBINED, and this is the most common case.

Command

```
:CONFigure:SRWireless:WLAC:SPECtrum:C80_80
```

Query

```
:CONFigure:SRWireless:WLAC:SPECtrum:C80_80?
```

Response

#	Description	Format	Units
1	802.11ac 80+80 separate or combined antenna setting	See definition	---

Parameters

#	Description	Format	Units	Min	Max	Default
1	802.11ac 80+80 separate or combined antenna setting	See definition	---	---	---	COMBINED

80+80 Separate or Combined setting:

SEPARATE	80+80 frequency segments on separate antenna ports.
COMBINED	80+80 frequency segments combined on one antenna port (or combined externally to the DUT).

Example of Use

```
:CONF:SRW:WLAC:SPEC:C80_80 SEPARATE        // The DUT uses separate antenna
                                           // ports for transmitting 80+80
                                           // frequency segments.
```

:CONFigure:SRWireless:WLAC:TCFLeakage:ULIMit

Function

Set or query the WLAN AC (OFDM) transmitter center frequency leakage upper limit.

Command

```
:CONFigure:SRWireless:WLAC:TCFLeakage:ULIMit
```

Note:

If any parameter is out of range, the command will fail and all the original parameter values will be retained.

Query

```
:CONFigure:SRWireless:WLAC:TCFLeakage:ULIMit?
```

Response

#	Description	Format	Units
1	Transmitter center frequency leakage upper limit (20 MHz channel)	Real number	dBm
2	Transmitter center frequency leakage upper limit (40 MHz channel)	Real number	dBm
3	Transmitter center frequency leakage upper limit (80 MHz channel)	Real number	dBm
4	Transmitter center frequency leakage upper limit (160 MHz channel)	Real number	dBm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Transmitter center frequency leakage upper limit (20 MHz channel)	Number+	dB	–100	100	–17.5	0.01 dB
2	Transmitter center frequency leakage upper limit (40 MHz channel)	Number+	dB	–100	100	–20.6	0.01 dB
3	Transmitter center frequency leakage upper limit (80 MHz channel)	Number+	dB	–100	100	–23.8	0.01 dB
4	Transmitter center frequency leakage upper limit (160 MHz channel)	Number+	dB	–100	100	–26.8	0.01 dB

Example of Use

```
:CONF:SRW:WLAC:TCFL:ULIM -17.5,-20.6,-23.8,-26.8
```

:CONFigure:SRWireless:WLAC:TCFTolerance:LIMit

Function

Set or query the WLAN AC (OFDM) transmitter center frequency tolerance limit (±).

Command

```
:CONFigure:SRWireless:WLAC:TCFTolerance:LIMit
```

Query

```
:CONFigure:SRWireless:WLAC:TCFTolerance:LIMit?
```

Response

#	Description	Format	Units
1	Transmitter center frequency tolerance limit (±)	Real number	ppm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Transmitter center frequency tolerance limit (±)	Number+	ppm	0	200	20	0.1 ppm

Example of Use

```
:CONF:SRW:WLAC:TCFT:LIM 19          // Set to ±19 ppm.
```

:CONFigure:SRWireless:WLAN:CCDF:GATE:ADD

Function

Add a gate to WLAN CCDF measurement.

Command

```
:CONFigure:SRWireless:WLAN:CCDF:GATE:ADD
```

Query

N/A

Response

N/A

Parameters

#	Description	Format	Units	Min	Max	Default
1	Gate number	Number	---	1	8	---

Example of Use

```
:CONF:SRW:WLAN:CCDF:GATE:ADD 2              // Add gate 2.
```

:CONFigure:SRWireless:WLAN:CCDF:GATE:REMove

Function

Remove a gate from WLAN CCDF measurement.

Command

```
:CONFigure:SRWireless:WLAN:CCDF:GATE:REMove
```

Query

N/A

Response

N/A

Parameters

#	Description	Format	Units	Min	Max	Default
1	Gate number	Number	---	1	8	---

Example of Use

```
:CONF:SRW:WLAN:CCDF:GATE:REM 1      // Remove gate 1
```

:CONFigure:SRWireless:WLAN:CCDF:GATE[:STATus]?

Function

Query gates assigned to WLAN CCDF measurement.

Command

N/A

Query

```
:CONFigure:SRWireless:WLAN:CCDF:GATE[:STATus]?
```

Response

#	Description	Format
1	Gate 1 assigned flag	Boolean
2	Gate 2 assigned flag	Boolean
3	Gate 3 assigned flag	Boolean
4	Gate 4 assigned flag	Boolean
5	Gate 5 assigned flag	Boolean
6	Gate 6 assigned flag	Boolean
7	Gate 7 assigned flag	Boolean
8	Gate 8 assigned flag	Boolean

Parameters

N/A

Example of Use

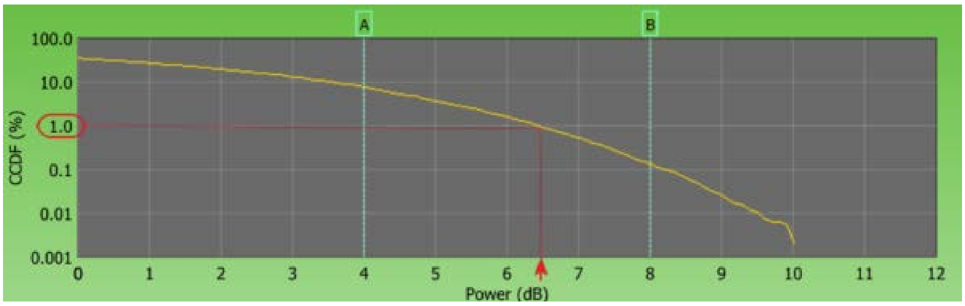
```
:CONF:SRW:WLAN:CCDF:GATE?            // Check which gates are assigned
                                     // to CCDF measurements.
```

> 1,1,0,0,0,0,0,0

:CONFigure:SRWireless:WLAN:CCDF:PSAMples

Function

Set a target 'percentage of samples' value (on the CCDF graph y-axis) for which the corresponding power distribution measurement will be returned by `:FETC:SRW:SUMM:WLAN:CCDF:PDIS?`



If the PSAMples value is set to 1.0%, as shown above, `:FETC:SRW:SUMM:WLAN:CCDF:PDIS?` will return the power distribution value corresponding to the point on the x-axis where the CCDF curve crosses the 1.0% line. In the above example this is approximately 6.5 dB.

Command

```
:CONFigure:SRWireless:WLAN:CCDF:PSAMples
```

Query

```
:CONFigure:SRWireless:WLAN:CCDF:PSAMples?
```

Response

#	Description	Format	Units	Resolution
1	Percentage of samples	Real number	%	0.001%

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Percentage of samples	Number+	%	0	100	1.000	0.001%

Example of Use

```
:CONF:SRW:WLAN:CCDF:PSAM 1.5
```

:CONFigure:SRWireless:WLAN:CMIMo:REFerence

Function

Set the Composite MIMO reference data for the active segment.

Note:

Before a composite MIMO measurement can be made, reference data must be transmitted to the instrument. The reference data is captured from the DUT operating in single stream mode and processed for use in composite MIMO measurements. The reference file contains information about the measurement set-up (MCS index, etc.) so that once the data is loaded the instrument is ready to perform the measurement without further configuration.

The data is transmitted in IEEE 488.2 <ARBITRARY BLOCK PROGRAM DATA> format.

Command

```
:CONFigure:SRWireless:WLAN:CMIMo:REFerence
```

Query

N/A

Response

N/A

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Composite MIMO reference data	Binary	---	---	---	---	----

Refer to 5.3.3 Binary_Header_Information.

Example of Use

```
// Step 1
// Open the file containing the binary reference data and read it
// into an array.

// Step 2
// Determine the length of the binary data (Example: 19284 bytes)

// Step 3
// Construct an IEEE488.2 <ARBITRARY BLOCK PROGRAM DATA> data header
//.string.  This takes the form "#mnn…n" - [Cross reference to 5.3.3.]
// (Example: #519284)
```

```
// Step 4
// Append the binary data to the header string and send the command.

:CONF:SRW:WLAN:CMIM:REF #519284<binary data - length 19284 bytes>
```

:CONFigure:SRWireless:WLAN:OBW:ULIMit

Function

Set or query the WLAN occupied bandwidth upper limit (for 99.9 % of transmitted power). Applies to all wireless LAN standards.

Command

```
:CONFigure:SRWireless:WLAN:OBW:ULIMit
```

Query

```
:CONFigure:SRWireless:WLAN:OBW:ULIMit?
```

Response

#	Description	Format	Units
1	Occupied bandwidth upper limit	Real number	Hz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Occupied bandwidth upper limit	Number+	Hz	0	200 MHz	200 MHz	1 Hz

Note:

Default value ensures that the limit test passes.

Example of Use

```
:CONF:SRW:WLAN:OBW:ULIM 12E+6       // Set limit to 12 MHz
```

:CONFigure:SRWireless:WLAN:POWer:GATE:ADD

Function

Add a gate to WLAN transmit power measurement.

Command

```
:CONFigure:SRWireless:WLAN:POWer:GATE:ADD
```

Query

N/A

Response

N/A

Parameters

#	Description	Format	Units	Min	Max	Default
1	Gate number	Number	---	1	8	---

Example of Use

```
:CONF:SRW:WLAN:POW:GATE:ADD 1       // Add gate 1
```

:CONFigure:SRWireless:WLAN:POWer:GATE:REMove

Function

Remove a gate from WLAN transmit power measurement.

Command

```
:CONFigure:SRWireless:WLAN:POWer:GATE:REMove
```

Query

N/A

Response

N/A

Parameters

#	Description	Format	Units	Min	Max	Default
1	Gate number	Number	---	1	8	---

Example of Use

```
:CONF:SRW:WLAN:POW:GATE:REM 1        // Remove gate 1
```

:CONFigure:SRWireless:WLAN:POWer:GATE[:STATus]?

Function

Query gates assigned to WLAN transmit power measurement.

Command

N/A

Query

```
:CONFigure:SRWireless:WLAN:POWer:GATE[:STATus]?
```

Response

#	Description	Format
1	Gate 1 assigned flag	Boolean
2	Gate 2 assigned flag	Boolean
3	Gate 3 assigned flag	Boolean
4	Gate 4 assigned flag	Boolean
5	Gate 5 assigned flag	Boolean
6	Gate 6 assigned flag	Boolean
7	Gate 7 assigned flag	Boolean
8	Gate 8 assigned flag	Boolean

Parameters

N/A

Example of Use

```
:CONF:SRW:WLAN:POW:GATE?            // Check which gates are assigned
                                    // to power measurements

> 1,0,0,0,0,0,0,0
```

:CONFigure:SRWireless:WLAN:PSD:ULIMit

Function

Set or query the WLAN power spectral density upper limit. (Applies to all wireless LAN standards.)

Command

```
:CONFigure:SRWireless:WLAN:PSD:ULIMit
```

Query

```
:CONFigure:SRWireless:WLAN:PSD:ULIMit?
```

Response

#	Description	Format	Units
1	Power spectral density upper limit	Real number	dBm / MHz

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Power spectral density upper limit	Number+	dBm / MHz	–100	100	20	0.01 dB

Example of Use

```
:CONF:SRW:WLAN:PSD:ULIM 25
```

:CONFigure:SRWireless:WLAN:SPECtrum:GATE:ADD

Function

Add a gate to WLAN spectrum measurement.

Command

```
:CONFigure:SRWireless:WLAN:SPECtrum:GATE:ADD
```

Query

N/A

Response

N/A

Parameters

#	Description	Format	Units	Min	Max	Default
1	Gate number	Number	---	1	8	---

Example of Use

```
:CONF:SRW:WLAN:SPEC:GATE:ADD 2            // Add gate 2
```

:CONFigure:SRWireless:WLAN:SPECtrum:GATE:REMove

Function

Remove a gate from WLAN spectrum measurement.

Command

```
:CONFigure:SRWireless:WLAN:SPECtrum:GATE:REMove
```

Query

N/A

Response

N/A

Parameters

#	Description	Format	Units	Min	Max	Default
1	Gate number	Number	---	1	8	---

Example of Use

```
:CONF:SRW:WLAN:SPEC:GATE:REM 2              // Remove gate 2
```

:CONFigure:SRWireless:WLAN:SPECtrum:GATE[:STATus]?

Function

Query gates assigned to WLAN spectrum measurement.

Command

N/A

Query

```
:CONFigure:SRWireless:WLAN:SPECtrum:GATE[:STATus]?
```

Response

#	Description	Format
1	Gate 1 assigned flag	Boolean
2	Gate 2 assigned flag	Boolean
3	Gate 3 assigned flag	Boolean
4	Gate 4 assigned flag	Boolean
5	Gate 5 assigned flag	Boolean
6	Gate 6 assigned flag	Boolean
7	Gate 7 assigned flag	Boolean
8	Gate 8 assigned flag	Boolean

Parameters

N/A

Example of Use

```
:CONF:SRW:WLAN:SPEC:GATE?             // Check which gates are assigned
                                      // to spectrum measurements.
> 1,0,0,0,0,0,0,0
```

:CONFigure:SRWireless:WLAN:SPOT:FREQuency:ADD

Function

Add a frequency offset to the list defined for use when fetching spot spectrum measurements. The list may contain up to 20 frequency offsets.

Notes:

The maximum frequency offset is equal to half the spectrum bandwidth. The spectrum bandwidth depends on the wireless standard.

If a frequency offset is found to be outside the range defined by the wireless standard when the measurement is initiated, the measurement will terminate with an execution error.

If wireless standard is AUTOOFDM or AUTODSSS and the frequency offset is found to be out of range, the results for that frequency offset will not be available.

Command

```
:CONFigure:SRWireless:WLAN:SPOT:FREQuency:ADD
```

Query

```
:CONFigure:SRWireless:WLAN:SPOT:FREQuency:ADD?
```

Response

#	Description	Format	Units	Resolution
1	Number of frequencies in list	Integer	---	---
2	First frequency	Integer	Hz	1 Hz
	…	…		
4	Last frequency	Integer	Hz	1 Hz

Parameters

#	Description	Format	Units	Min	Max	Def	Resolution
1	Frequency	Number	Hz	0	See notes	0	1 Hz

Example of Use

```
:CONF:SRW:WLAN:SPOT:FREQ:ADD 20MHZ
:CONF:SRW:WLAN:SPOT:FREQ:ADD 40E+06
:CONF:SRW:WLAN:SPOT:FREQ:ADD 53000000

:CONF:SRW:WLAN:SPOT:FREQ:ADD?

> 3,20000000,40000000,53000000
```

5

Remote Operation

:CONFigure:SRWireless:WLAN:SPOT:FREQuency:CLEar

Function

Clear the list of frequency offsets defined for use when fetching spot spectrum measurements. This command is required if you wish to change the list of defined frequency offsets between captures.

Command

```
:CONFigure:SRWireless:WLAN:SPOT:FREQuency:CLEar
```

Query

N/A

Response

N/A

Parameters

N/A

Example of Use

```
// Add frequencies to the list
:CONF:SRW:WLAN:SPOT:FREQ:ADD 20MHZ
:CONF:SRW:WLAN:SPOT:FREQ:ADD 40E+06
:CONF:SRW:WLAN:SPOT:FREQ:ADD 53000000

// Display the list
:CONF:SRW:WLAN:SPOT:FREQ:ADD?
> 3,20000000,40000000,53000000

// Clear the list and add another frequency.
:CONF:SRW:WLAN:SPOT:FREQ:CLE
:CONF:SRW:WLAN:SPOT:FREQ:ADD 15MHZ

// Display the list
:CONF:SRW:WLAN:SPOT:FREQ:ADD?
> 1,15000000
```

:CONFigure:SRWireless:WLAX:EVM:ULIMit

Function

Set or query the WLAN AX EVM upper limits

Command

```
:CONFigure:SRWireless:WLAX:EVM:ULIMit
```

Note:

If any parameter is out of range, the command will fail and all the original parameter values will be retained.

Query

```
:CONFigure:SRWireless:WLAX:EVM:ULIMit?
```

Response

#	Description	Format	Units
1	WLAN AX EVM upper limit (MCS 0)	Real number	dB
2	WLAN AX EVM upper limit (MCS 1)	Real number	dB
3	WLAN AX EVM upper limit (MCS 2)	Real number	dB
4	WLAN AX EVM upper limit (MCS 3)	Real number	dB
5	WLAN AX EVM upper limit (MCS 4)	Real number	dB
6	WLAN AX EVM upper limit (MCS 5)	Real number	dB
7	WLAN AX EVM upper limit (MCS 6)	Real number	dB
8	WLAN AX EVM upper limit (MCS 7)	Real number	dB
9	WLAN AX EVM upper limit (MCS 8)	Real number	dB
10	WLAN AX EVM upper limit (MCS 9)	Real number	dB
11	WLAN AX EVM upper limit (MCS 10)	Real number	dB
12	WLAN AX EVM upper limit (MCS 11)	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	WLAN AX EVM upper limit (MCS0)	Number+	dB	–100	100	–5	0.1 dB
2	WLAN AX EVM upper limit (MCS1)	Number+	dB	–100	100	–10	0.1 dB
3	WLAN AX EVM upper limit (MCS2)	Number+	dB	–100	100	–13	0.1 dB
4	WLAN AX EVM upper limit (MCS3)	Number+	dB	–100	100	–16	0.1 dB
5	WLAN AX EVM upper limit (MCS4)	Number+	dB	–100	100	–19	0.1 dB
6	WLAN AX EVM upper limit (MCS5)	Number+	dB	–100	100	–22	0.1 dB
7	WLAN AX EVM upper limit (MCS6)	Number+	dB	–100	100	–25	0.1 dB
8	WLAN AX EVM upper limit (MCS7)	Number+	dB	–100	100	–27	0.1 dB
9	WLAN AX EVM upper limit (MCS8)	Number+	dB	–100	100	–30	0.1 dB
10	WLAN AX EVM upper limit (MCS9)	Number+	dB	–100	100	–32	0.1 dB
11	WLAN AX EVM upper limit (MCS10)	Number+	dB	–100	100	–35	0.1 dB
12	WLAN AX EVM upper limit (MCS11)	Number+	dB	–100	100	–35	0.1 dB

Example of Use

```
:CONF:SRW:WLAX:EVM:ULIM -5,-10,-13,-16,-19,-22,-25,-27,-30,-32,-35,-35
```

:CONFigure:SRWireless:WLAX:GINTerval

Function

Set or query the guard interval of the expected WLAN AX packets in this segment.

Command

```
:CONFigure:SRWireless:WLAX:GINTerval
```

Query

```
:CONFigure:SRWireless:WLAX:GINTerval?
```

Response

#	Description	Format
1	WLAN AX Guard Interval	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN AX Guard Interval	See definition	---	---	---	0.8

WLAN AX Guard Interval:

0.8	0.8us
1.6	1.6us
3.2	3.2us

Example of Use

```
:CONF:SRW:WLAX:GINT 0.8
```

:CONFigure:SRWireless:WLAX:LSIZe

Function

Set or query the HE-LTF Size of the expected WLAN AX packets in this segment.

Command

```
:CONFigure:SRWireless:WLAX:LSIZe
```

Query

```
:CONFigure:SRWireless:WLAX:LSIZe?
```

Response

#	Description	Format
1	WLAN AX HE-LTF Size	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN AX HE-LTF Size	See definition	---	---	---	2

WLAN AX Guard Interval:

1	HE-LTF x1
2	HE-LTF x2
4	HE-LTF x4

Example of Use

```
:CONF:SRW:WLAX:LSIZ 1
```

:CONFigure:SRWireless:WLAX:MCS

Function

Set or query the expected MCS of the WLAN AX packets in this segment.

Command

```
:CONFigure:SRWireless:WLAX:MCS
```

Query

```
:CONFigure:SRWireless:WLAX:MCS?
```

Response

#	Description	Format
1	WLAN AX MCS	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN AX MCS	See definition	---	---	---	MCS11

WLAN AX MCS:

MCS0	MCS 0	(BPSK 1/2)
MCS1	MCS 1	(QPSK 1/2)
MCS2	MCS 2	(QPSK 3/4)
MCS3	MCS 3	(16-QAM 1/2)
MCS4	MCS 4	(16-QAM 3/4)
MCS5	MCS 5	(64-QAM 2/3)
MCS6	MCS 6	(64-QAM 3/4)
MCS7	MCS 7	(64-QAM 5/6)
MCS8	MCS 8	(256-QAM 3/4)
MCS9	MCS 9	(256-QAM 5/6)
MCS10	MCS 10	(1024-QAM 3/4)
MCS11	MCS 11	(1024-QAM 5/6)

Example of Use

```
:CONF:SRW:WLAX:MCS MCS11
```

:CONFigure:SRWireless:WLAX:PFORmat

Function

Set or query the expected PPDU Format of the WLAN AX packets in this segment.

Command

```
:CONFigure:SRWireless:WLAX:PFORmat
```

Query

```
:CONFigure:SRWireless:WLAX:PFORmat?
```

Response

#	Description	Format
1	WLAN AX PPDU Format	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN AX PPDU Format	See definition	---	---	---	SU

WLAN AX PPDU format:

SU	HE SU PPDU Format

Example of Use

```
:CONF:SRW:WLAX:PFOR SU
```

:CONFigure:SRWireless:WLAX:PTYPe

Function

Set or query the expected PPDU type of the WLAN AX packets in this segment.

Command

```
:CONFigure:SRWireless:WLAX:PTYPe
```

Query

```
:CONFigure:SRWireless:WLAX:PTYPe?
```

Response

#	Description	Format
1	WLAN AX PPDU Type	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN AX PPDU Type	See definition	---	---	---	HE80

WLAN AX PPDU Type:

HE20	20 MHz
HE40	40 MHz
HE80	80 MHz

Example of Use

```
:CONF:SRW:WLAX:PTYP HE80
```

:CONFigure:SRWireless:WLAX:SCFTolerance:LIMit

Function

Set or query the WLAN AX symbol clock frequency tolerance limit (±).

Command

```
:CONFigure:SRWireless:WLAX:SCFTolerance:LIMit
```

Query

```
:CONFigure:SRWireless:WLAX:SCFTolerance:LIMit?
```

Response

#	Description	Format	Units
	Symbol clock frequency tolerance limit (±)	Real number	ppm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Symbol clock frequency tolerance limit (±)	Number+	ppm	0	200	20	0.1 ppm

Example of Use

```
:CONF:SRW:WLAX:SCFT:LIM 19          // Set to ±19 ppm.
```

:CONFigure:SRWireless:WLAX:SFLatness:LIMit

Function

Set or query the WLAN AX spectral flatness limits

Command

`:CONFigure:SRWireless:WLAX:SFLatness:LIMit`

Note:

If any parameter is out of range, the command will fail and all the original parameter values will be retained.

Query

`:CONFigure:SRWireless:WLAX:SFLatness:LIMit?`

Response

#	Description	Format	Units
	20 MHz:		
1	Upper limit: subcarriers –122 to –85	Real number	dB
2	Lower limit: subcarriers –122 to –85	Real number	dB
3	Upper limit: subcarriers –84 to –2	Real number	dB
4	Lower limit: subcarriers –84 to –2	Real number	dB
5	Upper limit: subcarriers +2 to + 84	Real number	dB
6	Lower limit: subcarriers +2 to + 84	Real number	dB
7	Upper limit: subcarriers +85 to +122	Real number	dB
8	Lower limit: subcarriers +85 to +122	Real number	dB
	40 MHz:		
9	Upper limit: subcarriers –244 to –169	Real number	dB
10	Lower limit: subcarriers –244 to –169	Real number	dB
11	Upper limit: subcarriers –168 to –3	Real number	dB
12	Lower limit: subcarriers –168 to –3	Real number	dB
13	Upper limit: subcarriers +3 to +168	Real number	dB
14	Lower limit: subcarriers +3 to +168	Real number	dB
15	Upper limit: subcarriers +169 to +244	Real number	dB
16	Lower limit: subcarriers +169 to +244	Real number	dB
	80 MHz:		
17	Upper limit: subcarriers –500 to –345	Real number	dB
18	Lower limit: subcarriers –500 to –345	Real number	dB
19	Upper limit: subcarriers –344 to –3	Real number	dB
20	Lower limit: subcarriers –344 to –3	Real number	dB
21	Upper limit: subcarriers +3 to +344	Real number	dB
22	Lower limit: subcarriers +3 to +344	Real number	dB
23	Upper limit: subcarriers +345 to +500	Real number	dB
24	Lower limit: subcarriers +345 to +500	Real number	dB

	Reserved		
25	Reserved	Real number	dB
26	Reserved	Real number	dB
27	Reserved	Real number	dB
28	Reserved	Real number	dB
29	Reserved	Real number	dB
30	Reserved	Real number	dB
31	Reserved	Real number	dB
32	Reserved	Real number	dB
33	Reserved	Real number	dB
34	Reserved	Real number	dB
35	Reserved	Real number	dB
36	Reserved	Real number	dB
37	Reserved	Real number	dB
38	Reserved	Real number	dB
39	Reserved	Real number	dB
40	Reserved	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Def.	Resolution
	20 MHz:						
1	Upper limit: subcarriers –122 to –85	Number+	dB	–100	100	4	0.01 dB
2	Lower limit: subcarriers –122 to –85	Number+	dB	–100	100	–6	0.01 dB
3	Upper limit: subcarriers –84 to –2	Number+	dB	–100	100	4	0.01 dB
4	Lower limit: subcarriers –84 to –2	Number+	dB	–100	100	–4	0.01 dB
5	Upper limit: subcarriers +2 to + 84	Number+	dB	–100	100	4	0.01 dB
6	Lower limit: subcarriers +2 to + 84	Number+	dB	–100	100	–4	0.01 dB
7	Upper limit: subcarriers +85 to +122	Number+	dB	–100	100	4	0.01 dB
8	Lower limit: subcarriers +85 to +122	Number+	dB	–100	100	–6	0.01 dB
	40 MHz:						
9	Upper limit: subcarriers –244 to –169	Number+	dB	–100	100	4	0.01 dB
10	Lower limit: subcarriers –244 to –169	Number+	dB	–100	100	–6	0.01 dB
11	Upper limit: subcarriers –168 to –3	Number+	dB	–100	100	4	0.01 dB
12	Lower limit: subcarriers –168 to –3	Number+	dB	–100	100	–4	0.01 dB
13	Upper limit: subcarriers +3 to +168	Number+	dB	–100	100	4	0.01 dB
14	Lower limit: subcarriers +3 to +168	Number+	dB	–100	100	–4	0.01 dB
15	Upper limit: subcarriers +169 to +244	Number+	dB	–100	100	4	0.01 dB
16	Lower limit: subcarriers +169 to +244	Number+	dB	–100	100	–6	0.01 dB
	80 MHz:						
17	Upper limit: subcarriers –500 to –345	Number+	dB	–100	100	4	0.01 dB
18	Lower limit: subcarriers –500 to –345	Number+	dB	–100	100	–6	0.01 dB
19	Upper limit: subcarriers –344 to –3	Number+	dB	–100	100	4	0.01 dB
20	Lower limit: subcarriers –344 to –3	Number+	dB	–100	100	–4	0.01 dB
21	Upper limit: subcarriers +3 to +344	Number+	dB	–100	100	4	0.01 dB

22	Lower limit: subcarriers +3 to +344	Number+	dB	–100	100	–4	0.01 dB
23	Upper limit: subcarriers +345 to +500	Number+	dB	–100	100	4	0.01 dB
24	Lower limit: subcarriers +345 to +500	Number+	dB	–100	100	–6	0.01 dB
	Reserved:						
25	*Reserved*	Number+	dB	–100	100	4	0.01 dB
26	*Reserved*	Number+	dB	–100	100	–6	0.01 dB
27	*Reserved*	Number+	dB	–100	100	4	0.01 dB
28	*Reserved*	Number+	dB	–100	100	–4	0.01 dB
29	*Reserved*	Number+	dB	–100	100	4	0.01 dB
30	*Reserved*	Number+	dB	–100	100	–4	0.01 dB
31	*Reserved*	Number+	dB	–100	100	4	0.01 dB
32	*Reserved*	Number+	dB	–100	100	–6	0.01 dB
33	*Reserved*	Number+	dB	–100	100	4	0.01 dB
34	*Reserved*	Number+	dB	–100	100	–6	0.01 dB
35	*Reserved*	Number+	dB	–100	100	4	0.01 dB
36	*Reserved*	Number+	dB	–100	100	–4	0.01 dB
37	*Reserved*	Number+	dB	–100	100	4	0.01 dB
38	*Reserved*	Number+	dB	–100	100	–4	0.01 dB
39	*Reserved*	Number+	dB	–100	100	4	0.01 dB
40	*Reserved*	Number+	dB	–100	100	–6	0.01 dB

Example of Use

```
:CONF:SRW:WLAX:SFL:LIM 4,-6,4,-4,4,-4,4,-6,
                       4,-6,4,-4,4,-4,4,-6,
                       4,-6,4,-4,4,-4,4,-6,
                       4,-6,4,-4,4,-4,4,-6,4,-6,4,-4,4,-4,4,-6
```

:CONFigure:SRWireless:WLAX:SMASk:ULIMit

Function

Define the spectral mask limits for WLAN IEEE 802.11ax signals.

Command

```
:CONFigure:SRWireless:WLAX:SMASk:ULIMit
```

Query

```
:CONFigure:SRWireless:WLAX:SMASk:ULIMit
```

Response

#	Description	Format	Units	Resolution
1	fc – 30 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
2	fc – 20 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
3	fc – 10.5 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
4	fc + 10.5 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
5	fc + 20 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
6	fc + 30 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
7	fc – 60 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
8	fc – 40 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
9	fc – 20.5 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
10	fc + 20.5 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
11	fc + 40 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
12	fc + 60 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
13	fc – 120 MHz mask break point (80 MHz)	Real number	dBr	0.01 dB
14	fc – 80 MHz mask break point (80 MHz)	Real number	dBr	0.01 dB
15	fc – 40.5 MHz mask break point (80 MHz)	Real number	dBr	0.01 dB
16	fc + 40.5 MHz mask break point (80 MHz)	Real number	dBr	0.01 dB
17	fc + 80 MHz mask break point (80 MHz)	Real number	dBr	0.01 dB
18	fc + 120 MHz mask break point (80 MHz)	Real number	dBr	0.01 dB
19	Reserved	Real number	dBr	0.01 dB
20	Reserved	Real number	dBr	0.01 dB
21	Reserved	Real number	dBr	0.01 dB
22	Reserved	Real number	dBr	0.01 dB
23	Reserved	Real number	dBr	0.01 dB
24	Reserved	Real number	dBr	0.01 dB

Parameters

#	Description	Format	Units	Min	Max	Def	Res
1	fc – 30 MHz mask break point (20 MHz)	Number+	dBr	–100	0	–40	0.01 dB
2	fc – 20 MHz mask break point (20 MHz)	Number+	dBr	–100	0	–28	0.01 dB
3	fc – 10.5 MHz mask break point (20 MHz)	Number+	dBr	–100	0	–20	0.01 dB
4	fc + 10.5 MHz mask break point (20 MHz)	Number+	dBr	–100	0	–20	0.01 dB
5	fc + 20 MHz mask break point (20 MHz)	Number+	dBr	–100	0	–28	0.01 dB
6	fc + 30 MHz mask break point (20 MHz)	Number+	dBr	–100	0	–40	0.01 dB
7	fc – 60 MHz mask break point (40 MHz)	Number+	dBr	–100	0	–40	0.01 dB
8	fc – 40 MHz mask break point (40 MHz)	Number+	dBr	–100	0	–28	0.01 dB
9	fc – 20.5 MHz mask break point (40 MHz)	Number+	dBr	–100	0	–20	0.01 dB
10	fc + 20.5 MHz mask break point (40 MHz)	Number+	dBr	–100	0	–20	0.01 dB
11	fc + 40 MHz mask break point (40 MHz)	Number+	dBr	–100	0	–28	0.01 dB
12	fc + 60 MHz mask break point (40 MHz)	Number+	dBr	–100	0	–40	0.01 dB
13	fc – 120 MHz mask break point (80 MHz)	Number+	dBr	–100	0	–40	0.01 dB
14	fc – 80 MHz mask break point (80 MHz)	Number+	dBr	–100	0	–28	0.01 dB
15	fc – 40.5 MHz mask break point (80 MHz)	Number+	dBr	–100	0	–20	0.01 dB
16	fc + 40.5 MHz mask break point (80 MHz)	Number+	dBr	–100	0	–20	0.01 dB
17	fc + 80 MHz mask break point (80 MHz)	Number+	dBr	–100	0	–28	0.01 dB
18	fc + 120 MHz mask break point (80 MHz)	Number+	dBr	–100	0	–40	0.01 dB
19	Reserved	Number+	dBr	–100	0	–40	0.01 dB
20	Reserved	Number+	dBr	–100	0	–28	0.01 dB
21	Reserved	Number+	dBr	–100	0	–20	0.01 dB
22	Reserved	Number+	dBr	–100	0	–20	0.01 dB
23	Reserved	Number+	dBr	–100	0	–28	0.01 dB
24	Reserved	Number+	dBr	–100	0	–40	0.01 dB

Example of Use

```
:CONF:SRW:WLAX:SMAS:ULIM -40.00,-28.00,-20.00,-20.00,-28.00,-40.00,
                         -40.00,-28.00,-20.00,-20.00,-28.00,-40.00
                         -40.00,-28.00,-20.00,-20.00,-28.00,-40.00
                         nan,nan,nan,nan,nan,nan

:CONF:SRW:WLAX:SMAS:ULIM?
> -40.00,-28.00,-20.00,-20.00,-28.00,-40.00,
-40.00,-28.00,-20.00,-20.00,-28.00,-40.00
-40.00,-28.00,-20.00,-20.00,-28.00,-40.00
,nan,nan,nan,nan,nan,nan
```

:CONFigure:SRWireless:WLAX:TCFLeakage:ULIMit

Function

Set or query the WLAN AX transmitter center frequency leakage upper limit.

Command

```
:CONFigure:SRWireless:WLAX:TCFLeakage:ULIMit
```

Note:

If any parameter is out of range, the command will fail and all the original parameter values will be retained.

Query

```
:CONFigure:SRWireless:WLAX:TCFLeakage:ULIMit?
```

Response

#	Description	Format	Units
1	Transmitter center frequency leakage upper limit (20 MHz channel)	Real number	dBm
2	Transmitter center frequency leakage upper limit (40 MHz channel)	Real number	dBm
3	Transmitter center frequency leakage upper limit (80 MHz channel)	Real number	dBm
4	Reserved	Real number	dBm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Transmitter center frequency leakage upper limit (20 MHz channel)	Number+	dB	–100	100	–32.0	0.01 dB
2	Transmitter center frequency leakage upper limit (40 MHz channel)	Number+	dB	–100	100	–32.0	0.01 dB
3	Transmitter center frequency leakage upper limit (80 MHz channel)	Number+	dB	–100	100	–32.0	0.01 dB
4	Reserved	Number+	dB	–100	100	–32.0	0.01 dB

Example of Use

```
:CONF:SRW:WLAX:TCFL:ULIM -32.0,-32.0,-32.0,-32.0
```

:CONFigure:SRWireless:WLAX:TCFTolerance:LIMit

Function

Set or query the WLAN AX transmitter center frequency tolerance limit (±).

Command

```
:CONFigure:SRWireless:WLAX:TCFTolerance:LIMit
```

Query

```
:CONFigure:SRWireless:WLAX:TCFTolerance:LIMit?
```

Response

#	Description	Format	Units
1	Transmitter center frequency tolerance limit (±)	Real number	ppm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Transmitter center frequency tolerance limit (±)	Number+	ppm	0	200	20	0.1 ppm

Example of Use

```
:CONF:SRW:WLAX:TCFT:LIM 19          // Set to ±19 ppm.
```

:CONFigure:SRWireless:WLB:CCFTolerance:ULIMit

Function

Set or query the WLAN B (DSSS) chip clock frequency tolerance limit (±).

Command

```
:CONFigure:SRWireless:WLB:CCFTolerance:ULIMit
```

Query

```
:CONFigure:SRWireless:WLB:CCFTolerance:ULIMit?
```

Response

#	Description	Format	Units
1	Chip clock frequency tolerance limit (±)	Real number	ppm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Chip clock frequency tolerance limit (±)	Number+	ppm	0	200	25	0.1 ppm

Example of Use

```
:CONF:SRW:WLB:CCFT:ULIM 30          // Set to ±30 ppm.
```

:CONFigure:SRWireless:WLB:DRATe

Function

Set or query the expected data rate of WLAN B packets in this segment.

Command

```
:CONFigure:SRWireless:WLB:DRATe
```

Query

```
:CONFigure:SRWireless:WLB:DRATe?
```

Response

#	Description	Format
1	WLAN B data rate	See definition

Parameters

#	Description	Format	Units	Default
1	WLAN B data rate	See definition	---	11MBPS

WLAN B data rate:

1MBPS	1 Mbps
2MBPS	2 Mbps
5_5MBPS	5.5 Mbps
11MBPS	11 Mbps

Example of Use

```
:CONF:SRW:WLB:DRAT 5_5MBPS
```

:CONFigure:SRWireless:WLB:EVM:ULIMit

Function

Set or query the WLAN B EVM upper limits.

Command

```
:CONFigure:SRWireless:WLB:EVM:ULIMit
```

Query

```
:CONFigure:SRWireless:WLB:EVM:ULIMit?
```

Response

#	Description	Format	Units
1	WLAN B EVM upper limit (1 Mbps)	Real number	dB
2	WLAN B EVM upper limit (2 Mbps)	Real number	dB
3	WLAN B EVM upper limit (5.5 Mbps)	Real number	dB
4	WLAN B EVM upper limit (11 Mbps)	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	WLAN B EVM upper limit (1 Mbps)	Number+	dB	–60	0	–9.12	0.01
2	WLAN B EVM upper limit (2 Mbps)	Number+	dB	–60	0	–9.12	0.01
3	WLAN B EVM upper limit (5.5 Mbps)	Number+	dB	–60	0	–9.12	0.01
4	WLAN B EVM upper limit (11 Mbps)	Number+	dB	–60	0	–9.12	0.01

Example of Use

```
:CONF:SRW:WLB:EVM:ULIM -10,-10,-10,-10
```

:CONFigure:SRWireless:WLB:FALPha

Function

Set or query the root raised cosine filter alpha (applied if root raised cosine filter selected).

Command

```
:CONFigure:SRWireless:WLB:FALPha
```

Query

```
:CONFigure:SRWireless:WLB:FALPha?
```

Response

#	Description	Format
1	WLAN B filter alpha (RRC filter)	Real number

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	WLAN B filter alpha (RRC filter)	Number+	---	0.3	1	0.35	0.01

Example of Use

```
:CONF:SRW:WLB:FTYP RRC       // Set the filter type.
:CONF:SRW:WLB:FALP 0.4       // Set the filter alpha value.
```

:CONFigure:SRWireless:WLB:FBT

Function

Set or query the gaussian filter BT (applied if gaussian filter selected).

Command

```
:CONFigure:SRWireless:WLB:FBT
```

Query

```
:CONFigure:SRWireless:WLB:FBT?
```

Response

#	Description	Format	Units
	WLAN B filter BT (Gaussian filter)	Real number	---

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	WLAN B filter BT (Gaussian filter)	Number+	---	0.3	1	0.5	0.1

Example of Use

```
:CONF:SRW:WLB:FTYP GAUSSIAN        // Set the filter type.
:CONF:SRW:WLB:FBT 0.4              // Set the BT value.
```

:CONFigure:SRWireless:WLB:FTYPe

Function

Set or query the expected filter type.

Command

```
:CONFigure:SRWireless:WLB:FTYPe
```

Query

```
:CONFigure:SRWireless:WLB:FTYPe?
```

Response

#	Description	Format
1	WLAN B filter type	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN B filter type	See definition	---	---	---	NONE

WLAN B filter type:

GAUSSIAN	Gaussian
RRC	Root raised cosine
NONE	No filter

Example of Use

```
:CONF:SRW:WLB:FTYP GAUSSIAN
```

:CONFigure:SRWireless:WLB:IQCSuppress:LLIMit

Function

WLAN B (DSSS) RF carrier suppression (from IQ Offset) lower limit.
This function allows the limit for RF carrier suppression to be set for the case when it is calculated from the IQ Offset. (There is an independent limit setting for the case when carrier suppression is calculated from the spectrum.
See :CONFigure:SRWireless:WLB:RFCSuppress:LLIMit.)

Command

```
:CONFigure:SRWireless:WLB:IQCSuppress:LLIMit
```

Query

```
:CONFigure:SRWireless:WLB:IQCSuppress:LLIMit?
```

Response

#	Description	Format	Units
1	RF carrier suppression (from IQ offset) lower limit	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	RF carrier suppression (from IQ offset) lower limit	Number+	dB	0	100	15	0.01 dB

Example of Use

```
:CONF:SRW:WLB:IQCS:LLIM 20
```

:CONFigure:SRWireless:WLB:RFCSuppress:LLIMit

Function

Set or query the WLAN B (DSSS) RF carrier suppression lower limit.

Command

```
:CONFigure:SRWireless:WLB:RFCSuppress:LLIMit
```

Query

```
:CONFigure:SRWireless:WLB:RFCSuppress:LLIMit?
```

Response

#	Description	Format	Units
1	RF carrier suppression lower limit	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	RF carrier suppression lower limit	Number+	dB	0	100	15	0.01 dB

Example of Use

```
:CONF:SRW:WLB:RFCS:LLIM 20
```

:CONFigure:SRWireless:WLB:SMASk

Function

Set the standard used for 802.11b spectral mask measurements.

It can be set to either IEEE 802.11 or ARIB RCR-STD 33 (5.0).

Spectral profile and mask limits will follow the specifications of the selected standard.

Note:

ARIB RCR-STD 33 (5.0) option is intended for 802.11b channel 14 (2484 MHz) as described at “16.4.7.5 Transmit spectrum mask” in IEEE 802.11 Standard. Channel 14 is used only in Japan.

Command

```
:CONFigure:SRWireless:WLB:SMASk
```

Query

```
:CONFigure:SRWireless:WLB:SMASk?
```

Response

#	Description	Format	Units	Resolution
1	Standard	See definition	---	---

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Standard	See definition	---	---	---	IEEE	---

Standard:

IEEE	IEEE Standard.
ARIBSTD33	ARIB RCR-STD 33 (5.0) Standard.

Example of Use

```
:CONF:SRW:WLB:SMAS ARIBSTD33        // Select ARIB RCR-STD 33 (5.0) Standard.
```

:CONFigure:SRWireless:WLB:SMASk:ULIMit

Function

Define the spectral mask limits for WLAN IEEE 802.11b signals.

Command

```
:CONFigure:SRWireless:WLB:SMASk:ULIMit
```

Query

```
:CONFigure:SRWireless:WLB:SMASk:ULIMit?
```

Response

#	Description	Format	Units	Resolution
1	fc – 22 MHz mask break point	Real number	dBr	0.01 dB
2	fc – 11 MHz mask break point	Real number	dBr	0.01 dB
3	fc + 11 MHz mask break point	Real number	dBr	0.01 dB
4	fc + 22 MHz mask break point	Real number	dBr	0.01 dB

Parameters

#	Description	Format	Units	Min	Max	Def	Resolution
1	fc – 22 MHz mask break point	Number+	dBr	–100	0	–50	0.01 dB
2	fc – 11 MHz mask break point	Number+	dBr	–100	0	–30	0.01 dB
3	fc + 11 MHz mask break point	Number+	dBr	–100	0	–30	0.01 dB
4	fc + 22 MHz mask break point	Number+	dBr	–100	0	–50	0.01 dB

Example of Use

```
:CONF:SRW:WLB:SMAS:ULIM -50.00,-30.00,-30.00,-50.00
:CONF:SRW:WLB:SMAS:ULIM?
> -50.00,-30.00,-30.00,-50.00
```

5

Remote Operation

:CONFigure:SRWireless:WLB:TCFTolerance:LIMit

Function

Set or query the WLAN B (DSSS) transmitter center frequency tolerance limit (±).

Command

```
:CONFigure:SRWireless:WLB:TCFTolerance:LIMit
```

Query

```
:CONFigure:SRWireless:WLB:TCFTolerance:LIMit?
```

Response

#	Description	Format	Units
1	Transmitter center frequency tolerance limit (±)	Real number	ppm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Transmitter center frequency tolerance limit (±)	Number +	ppm	0	200	25	0.1 ppm

Example of Use

```
:CONF:SRW:WLB:TCFT:LIM 24          // Set to ±24 ppm.
```

:CONFigure:SRWireless:WLG:DRATe

Function

Set or query the expected data rate of WLAN G packets in this segment.

Command

```
:CONFigure:SRWireless:WLG:DRATe
```

Query

```
:CONFigure:SRWireless:WLG:DRATe?
```

Response

#	Description	Format
1	WLAN G data rate	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN G data rate	See definition	---	---	---	54MBPS

WLAN A data rate:

6MBPS	6 Mbps
9MBPS	9 Mbps
12MBPS	12 Mbps
18MBPS	18 Mbps
24MBPS	24 Mbps
36MBPS	36 Mbps
48MBPS	48 Mbps
54MBPS	54 Mbps

Example of Use

```
:CONF:SRW:WLG:DRAT 6MBPS
```

:CONFigure:SRWireless:WLG:EVM:ULIMit

Function

Set or query the WLAN G EVM upper limits

Command

```
:CONFigure:SRWireless:WLG:EVM:ULIMit
```

Note:

If any parameter is out of range, the command will fail and all the original parameter values will be retained.

Query

```
:CONFigure:SRWireless:WLG:EVM:ULIMit?
```

Response

#	Description	Format	Units
1	WLAN G EVM upper limit (6 Mbps)	Real number	dB
2	WLAN G EVM upper limit (9 Mbps)	Real number	dB
3	WLAN G EVM upper limit (12 Mbps)	Real number	dB
4	WLAN G EVM upper limit (18 Mbps)	Real number	dB
5	WLAN G EVM upper limit (24 Mbps)	Real number	dB
6	WLAN G EVM upper limit (36 Mbps)	Real number	dB
7	WLAN G EVM upper limit (48 Mbps)	Real number	dB
8	WLAN G EVM upper limit (54 Mbps)	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	WLAN G EVM upper limit (6 Mbps)	Number+	dB	–100	100	–5	0.1 dB
2	WLAN G EVM upper limit (9 Mbps)	Number+	dB	–100	100	–8	0.1 dB
3	WLAN G EVM upper limit (12 Mbps)	Number+	dB	–100	100	–10	0.1 dB
4	WLAN G EVM upper limit (18 Mbps)	Number+	dB	–100	100	–13	0.1 dB
5	WLAN G EVM upper limit (24 Mbps)	Number+	dB	–100	100	–16	0.1 dB
6	WLAN G EVM upper limit (36 Mbps)	Number+	dB	–100	100	–19	0.1 dB
7	WLAN G EVM upper limit (48 Mbps)	Number+	dB	–100	100	–22	0.1 dB
8	WLAN G EVM upper limit (54 Mbps)	Number+	dB	–100	100	–25	0.1 dB

Example of Use

```
:CONF:SRW:WLG:EVM:ULIM -5,-8,-10,-13,-16,-19,-22,-25
```

:CONFigure:SRWireless:WLG:SCFTolerance:LIMit

Function

Set or query the WLAN G (OFDM) symbol clock frequency tolerance limit (±).

Command

```
:CONFigure:SRWireless:WLG:SCFTolerance:LIMit
```

Query

```
:CONFigure:SRWireless:WLG:SCFTolerance:LIMit?
```

Response

#	Description	Format	Units
1	Symbol clock frequency tolerance limit (±)	Real number	ppm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Symbol clock frequency tolerance limit (±)	Number+	ppm	0	200	25	0.1 ppm

Example of Use

```
:CONF:SRW:WLG:SCFT:LIM 20          // Set to ±20 ppm.
```

:CONFigure:SRWireless:WLG:SFLatness:LIMit

Function

Set or query the WLAN G (OFDM) spectral flatness limits.

Command

```
:CONFigure:SRWireless:WLG:SFLatness:LIMit
```

Note:

If any parameter is out of range, the command will fail and all the original parameter values will be retained.

Query

```
:CONFigure:SRWireless:WLG:SFLatness:LIMit?
```

Response

#	Description	Format	Units
1	Upper limit: subcarriers –26 to –17	Real number	dB
2	Lower limit: subcarriers –26 to –17	Real number	dB
3	Upper limit: subcarriers –16 to –1	Real number	dB
4	Lower limit: subcarriers –16 to –1	Real number	dB
5	Upper limit: subcarriers +1 to +16	Real number	dB
6	Lower limit: subcarriers +1 to +16	Real number	dB
7	Upper limit: subcarriers +17 to +26	Real number	dB
8	Lower limit: subcarriers +17 to +26	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Upper limit: subcarriers –26 to –17	Number+	dB	–100	100	2	0.01 dB
2	Lower limit: subcarriers –26 to –17	Number+	dB	–100	100	–4	0.01 dB
3	Upper limit: subcarriers –16 to – 1	Number+	dB	–100	100	2	0.01 dB
4	Lower limit: subcarriers –16 to – 1	Number+	dB	–100	100	–2	0.01 dB
5	Upper limit: subcarriers +1 to +16	Number+	dB	–100	100	2	0.01 dB
6	Lower limit: subcarriers +1 to +16	Number+	dB	–100	100	–2	0.01 dB
7	Upper limit: subcarriers +17 to +26	Number+	dB	–100	100	2	0.01 dB
8	Lower limit: subcarriers +17 to +26	Number+	dB	–100	100	–4	0.01 dB

Example of Use

```
:CONF:SRW:WLG:SFL:LIM 2,-5,2,-2,2,-2,2,-5
```

:CONFigure:SRWireless:WLG:SMASk:ULIMit

Function

Define the spectral mask limits for WLAN IEEE 802.11g signals.

Command

```
:CONFigure:SRWireless:WLG:SMASk:ULIMit
```

Query

```
:CONFigure:SRWireless:WLG:SMASk:ULIMit?
```

Response

#	Description	Format	Units	Resolution
1	fc – 30 MHz mask break point	Real number	dBr	0.01 dB
2	fc – 20 MHz mask break point	Real number	dBr	0.01 dB
3	fc – 11 MHz mask break point	Real number	dBr	0.01 dB
4	fc + 11 MHz mask break point	Real number	dBr	0.01 dB
5	fc + 20 MHz mask break point	Real number	dBr	0.01 dB
6	fc + 30 MHz mask break point	Real number	dBr	0.01 dB

Parameters

#	Description	Format	Units	Min	Max	Def	Resolution
1	fc – 30 MHz mask break point	Number+	dBr	–100	0	–40	0.01 dB
2	fc – 20 MHz mask break point	Number+	dBr	–100	0	–28	0.01 dB
3	fc – 11 MHz mask break point	Number+	dBr	–100	0	–20	0.01 dB
4	fc + 11 MHz mask break point	Number+	dBr	–100	0	–20	0.01 dB
5	fc + 20 MHz mask break point	Number+	dBr	–100	0	–28	0.01 dB
6	fc + 30 MHz mask break point	Number+	dBr	–100	0	–40	0.01 dB

Example of Use

```
:CONF:SRW:WLG:SMAS:ULIM -40.00,-28.00,-20.00,-20.00,-28.00,-40.00
:CONF:SRW:WLG:SMAS:ULIM
> -40.00,-28.00,-20.00,-20.00,-28.00,-40.00
```

5

Remote Operation

:CONFigure:SRWireless:WLG:TCFLeakage:ULIMit

Function

Set or query the WLAN G (OFDM) transmitter center frequency leakage upper limit.

Command

```
:CONFigure:SRWireless:WLG:TCFLeakage:ULIMit
```

Query

```
:CONFigure:SRWireless:WLG:TCFLeakage:ULIMit?
```

Response

#	Description	Format	Units
1	Transmitter center frequency leakage upper limit	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Transmitter center frequency leakage upper limit	Number+	dB	–100	100	–15	0.01 dB

Example of Use

```
:CONF:SRW:WLG:TCFL:ULIM -14
```

:CONFigure:SRWireless:WLG:TCFTolerance:LIMit

Function

Set or query the WLAN G (OFDM) transmitter center frequency tolerance limit (±).

Command

```
:CONFigure:SRWireless:WLG:TCFTolerance:LIMit
```

Query

```
:CONFigure:SRWireless:WLG:TCFTolerance:LIMit?
```

Response

#	Description	Format	Units
1	Transmitter center frequency tolerance limit (±)	Real number	ppm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Transmitter center frequency tolerance limit (±)	Number+	ppm	0	200	25	0.1 ppm

Example of Use

```
:CONF:SRW:WLG:TCFT:LIM 24          // Set to ±24 ppm.
```

:CONFigure:SRWireless:WLN:EVM:ULIMit

Function

Set or query the WLAN N EVM upper limits

Command

```
:CONFigure:SRWireless:WLN:EVM:ULIMit
```

Query

```
:CONFigure:SRWireless:WLN:EVM:ULIMit?
```

Response

#	Description	Format	Units
1	WLAN N EVM upper limit (MCS 0)	Real number	dB
2	WLAN N EVM upper limit (MCS 1)	Real number	dB
3	WLAN N EVM upper limit (MCS 2)	Real number	dB
4	WLAN N EVM upper limit (MCS 3)	Real number	dB
5	WLAN N EVM upper limit (MCS 4)	Real number	dB
6	WLAN N EVM upper limit (MCS 5)	Real number	dB
7	WLAN N EVM upper limit (MCS 6)	Real number	dB
8	WLAN N EVM upper limit (MCS 7)	Real number	dB
9	WLAN N EVM upper limit (MCS 32)	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	WLAN N EVM upper limit (MCS 0)	Number+	dB	–100	100	–5	0.1 dB
2	WLAN N EVM upper limit (MCS 1)	Number+	dB	–100	100	–10	0.1 dB
3	WLAN N EVM upper limit (MCS 2)	Number+	dB	–100	100	–13	0.1 dB
4	WLAN N EVM upper limit (MCS 3)	Number+	dB	–100	100	–16	0.1 dB
5	WLAN N EVM upper limit (MCS 4)	Number+	dB	–100	100	–19	0.1 dB
6	WLAN N EVM upper limit (MCS 5)	Number+	dB	–100	100	–22	0.1 dB
7	WLAN N EVM upper limit (MCS 6)	Number+	dB	–100	100	–25	0.1 dB
8	WLAN N EVM upper limit (MCS 7)	Number+	dB	–100	100	–28	0.1 dB
9	WLAN N EVM upper limit (MCS 32)	Number+	dB	–100	100	–5	0.1 dB

Example of Use

```
:CONF:SRW:WLN:EVM:ULIM -5,-10,-13,-16,-19,-22,-25,-28,-5
```

:CONFigure:SRWireless:WLN:GINTerval

Function

Set or query the expected guard interval of the WLAN N packets in this segment.

Note:

For SISO signals, this setting only applies when the PPDU format is HT Mixed Mode. For HT Greenfield, the guard interval is always LONG.

For MIMO signals, the guard interval may be LONG or SHORT, regardless of the PPDU format.

Command

```
:CONFigure:SRWireless:WLN:GINTerval
```

Query

```
:CONFigure:SRWireless:WLN:GINTerval?
```

Response

#	Description	Format
1	WLAN N Guard Interval	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN N Guard Interval	See definition	---	---	---	LONG

WLAN N Guard Interval:

LONG	Long guard interval (800 ns)
SHORT	Short guard interval (400 ns)

Example of Use

```
:CONF:SRW:WLN:PFOR MIXED    // Guard interval setting is only relevant
:CONF:SRW:WLN:GINT SHORT    // when the PPDU format is HT Mixed Mode.
```

5

Remote Operation

:CONFigure:SRWireless:WLN:MCS

Function

Set or query the expected MCS number of the WLAN N packets in this segment.

Command

```
:CONFigure:SRWireless:WLN:MCS
```

Query

```
:CONFigure:SRWireless:WLN:MCS?
```

Response

#	Description	Format
1	WLAN N MCS	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN N MCS	See definition	---	---	---	MCS7

WLAN N MCS:

MCS0	MCS 0	(BPSK)
MCS1	MCS 1	(QPSK)
MCS2	MCS 2	(QPSK)
MCS3	MCS 3	(16QAM)
MCS4	MCS 4	(16QAM)
MCS5	MCS 5	(64QAM)
MCS6	MCS 6	(64QAM)
MCS7	MCS 7	(64QAM)
MCS32	MCS 32	(BPSK)

Example of Use

```
:CONF:SRW:WLN:MCS MCS3
```

:CONFigure:SRWireless:WLN:PFORmat

Function

Set or query the expected PPDU format of the WLAN N packets in this segment.

Command

```
:CONFigure:SRWireless:WLN:PFORmat
```

Query

```
:CONFigure:SRWireless:WLN:PFORmat?
```

Response

#	Description	Format
1	WLAN N PPDU Format	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN N PPDU Format	See definition	---	---	---	MIXED

WLAN N PPDU Format:

MIXED	HT Mixed
GREENFIELD	HT Greenfield

Example of Use

```
:CONF:SRW:WLN:PFOR GREENFIELD
```

:CONFigure:SRWireless:WLN:PTYPe

Function

Set or query the expected PPDU type of the WLAN N packets in this segment.

Command

```
:CONFigure:SRWireless:WLN:PTYPe
```

Query

```
:CONFigure:SRWireless:WLN:PTYPe?
```

Response

#	Description	Format
1	WLAN N PPDU Type	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN N PPDU Type	See definition	---	---	---	M40

WLAN N PPDU Type:

M20	Data is transmitted over a signal with 20 MHz bandwidth
M40	Data is transmitted over two adjacent signals with 20 MHz bandwidth
M40UPPER	Data is transmitted over a signal with 20 MHz bandwidth in the upper sideband of a 40 MHz channel
M40LOWER	Data is transmitted over a signal with 20 MHz bandwidth in the lower sideband of a 40 MHz channel.
M40DUP	Data is transmitted twice using two adjacent signals with 20 MHz bandwidth

Example of Use

```
:CONF:SRW:WLN:PTYP M40
```

:CONFigure:SRWireless:WLN:SCFTolerance:LIMit

Function

Set or query the WLAN N (OFDM) symbol clock frequency tolerance limit (±).

Command

```
:CONFigure:SRWireless:WLN:SCFTolerance:LIMit
```

Query

```
:CONFigure:SRWireless:WLN:SCFTolerance:LIMit?
```

Response

#	Description	Format	Units
1	Symbol clock frequency tolerance limit (±) 2.4 GHz band	Real number	ppm
2	Symbol clock frequency tolerance limit (±) 5 GHz band	Real number	ppm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Symbol clock frequency tolerance limit (±) 2.4 GHz band	Number+	ppm	0	200	25	0.1 ppm
2	Symbol clock frequency tolerance limit (±) 5 GHz band	Number+	ppm	0	200	20	0.1 ppm

Example of Use

```
:CONF:SRW:WLN:SCFT:LIM 24,19// Set to ±24 ppm for 2.4 GHz band
                                 // and ±19 ppm for 5 GHz band.
```

:CONFigure:SRWireless:WLN:SFLatness:LIMit

Function

Set or query the WLAN N (OFDM) spectral flatness limits.

Command

```
:CONFigure:SRWireless:WLN:SFLatness:LIMit
```

Note:

If any parameter is out of range, the command will fail and all the original parameter values will be retained.

Query

```
:CONFigure:SRWireless:WLN:SFLatness:LIMit?
```

Response

#	Description	Format	Units
	20 MHz:		
1	Upper limit: subcarriers –28 to –17	Real number	dB
2	Lower limit: subcarriers –28 to –17	Real number	dB
3	Upper limit: subcarriers –16 to –1	Real number	dB
4	Lower limit: subcarriers –16 to –1	Real number	dB
5	Upper limit: subcarriers +1 to +16	Real number	dB
6	Lower limit: subcarriers +1 to +16	Real number	dB
7	Upper limit: subcarriers +17 to +28	Real number	dB
8	Lower limit: subcarriers +17 to +28	Real number	dB
	40 MHz:		
9	Upper limit: subcarriers –58 to –43	Real number	dB
10	Lower limit: subcarriers –58 to –43	Real number	dB
11	Upper limit: subcarriers –42 to –2	Real number	dB
12	Lower limit: subcarriers –42 to –2	Real number	dB
13	Upper limit: subcarriers +2 to +42	Real number	dB
14	Lower limit: subcarriers +2 to +42	Real number	dB
15	Upper limit: subcarriers +43 to +48	Real number	dB
16	Lower limit: subcarriers +43 to +48	Real number	dB
	40 MHz (MCS 32):		
17	Upper limit: subcarriers –58 to –43	Real number	dB
18	Lower limit: subcarriers –58 to –43	Real number	dB
19	Upper limit: subcarriers –42 to –33	Real number	dB
20	Lower limit: subcarriers –42 to –33	Real number	dB
21	Upper limit: subcarriers –31 to –6	Real number	dB
22	Lower limit: subcarriers –31 to –6	Real number	dB
23	Lower limit: subcarriers +6 to +31	Real number	dB

24	Upper limit: subcarriers +6 to +31	Real number	dB
25	Lower limit: subcarriers +33 to +42	Real number	dB
26	Lower limit: subcarriers +33 to +42	Real number	dB
27	Upper limit: subcarriers +43 to +58	Real number	dB
28	Upper limit: subcarriers +43 to +58	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Def.	Resolution
	20 MHz:						
1	Upper limit: subcarriers –28 to –17	Number+	dB	–100	100	2	0.01 dB
2	Lower limit: subcarriers –28 to –17	Number+	dB	–100	100	–4	0.01 dB
3	Upper limit: subcarriers –16 to –1	Number+	dB	–100	100	2	0.01 dB
4	Lower limit: subcarriers –16 to –1	Number+	dB	–100	100	–2	0.01 dB
5	Upper limit: subcarriers +1 to +16	Number+	dB	–100	100	2	0.01 dB
6	Lower limit: subcarriers +1 to +16	Number+	dB	–100	100	–2	0.01 dB
7	Upper limit: subcarriers +17 to +28	Number+	dB	–100	100	2	0.01 dB
8	Lower limit: subcarriers +17 to +28	Number+	dB	–100	100	–4	0.01 dB
	40 MHz:						
9	Upper limit: subcarriers –58 to –43	Number+	dB	–100	100	2	0.01 dB
10	Lower limit: subcarriers –58 to –43	Number+	dB	–100	100	–4	0.01 dB
11	Upper limit: subcarriers –42 to –2	Number+	dB	–100	100	2	0.01 dB
12	Lower limit: subcarriers –42 to –2	Number+	dB	–100	100	–2	0.01 dB
13	Upper limit: subcarriers +2 to +42	Number+	dB	–100	100	2	0.01 dB
14	Lower limit: subcarriers +2 to +42	Number+	dB	–100	100	–2	0.01 dB
15	Upper limit: subcarriers +43 to +58	Number+	dB	–100	100	2	0.01 dB
16	Lower limit: subcarriers +43 to +58	Number+	dB	–100	100	–4	0.01 dB
	40 MHz (MCS 32):						
17	Upper limit: subcarriers –58 to –43	Number+	dB	–100	100	2	0.01 dB
18	Lower limit: subcarriers –58 to –43	Number+	dB	–100	100	–4	0.01 dB
19	Upper limit: subcarriers –42 to –33	Number+	dB	–100	100	2	0.01 dB
20	Lower limit: subcarriers –42 to –33	Number+	dB	–100	100	–2	0.01 dB
21	Upper limit: subcarriers –31 to –6	Number+	dB	–100	100	2	0.01 dB
22	Lower limit: subcarriers –32 to –6	Number+	dB	–100	100	–2	0.01 dB
23	Lower limit: subcarriers +6 to +31	Number+	dB	–100	100	2	0.01 dB
24	Upper limit: subcarriers +6 to +31	Number+	dB	–100	100	–2	0.01 dB
25	Lower limit: subcarriers +33 to +42	Number+	dB	–100	100	2	0.01 dB
26	Lower limit: subcarriers +33 to +42	Number+	dB	–100	100	–2	0.01 dB
27	Upper limit: subcarriers +43 to +58	Number+	dB	–100	100	2	0.01 dB
28	Upper limit: subcarriers +43 to +58	Number+	dB	–100	100	–4	0.01 dB

Example of Use

```
:CONF:SRW:WLN:SFL:LIM 2,-5,2,-2,2,-2,2,-5,
                      2,-4,2,-2,2,-2,2,-4,
                      2,-4,2,-2,2,-2,2,-2,2,-2,2,-4
```

5

Remote Operation

:CONFigure:SRWireless:WLN:SMASk:ULIMit

Function

Define the spectral mask limits for WLAN IEEE 802.11n signals.

Command

```
:CONFigure:SRWireless:WLN:SMASk:ULIMit
```

Query

```
:CONFigure:SRWireless:WLN:SMASk:ULIMit
```

Response

#	Description	Format	Units	Resolution
1	fc – 30 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
2	fc – 20 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
3	fc – 11 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
4	fc + 11 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
5	fc + 20 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
6	fc + 30 MHz mask break point (20 MHz)	Real number	dBr	0.01 dB
7	fc – 60 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
8	fc – 40 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
9	fc – 21 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
10	fc + 21 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
11	fc + 40 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB
12	fc + 60 MHz mask break point (40 MHz)	Real number	dBr	0.01 dB

Parameters

#	Description	Format	Units	Min	Max	Def	Res
1	fc – 30 MHz mask break point (20 MHz)	Number+	dBr	–100	0	–45	0.01 dB
2	fc – 20 MHz mask break point (20 MHz)	Number+	dBr	–100	0	–28	0.01 dB
3	fc – 11 MHz mask break point (20 MHz)	Number+	dBr	–100	0	–20	0.01 dB
4	fc + 11 MHz mask break point (20 MHz)	Number+	dBr	–100	0	–20	0.01 dB
5	fc + 20 MHz mask break point (20 MHz)	Number+	dBr	–100	0	–28	0.01 dB
6	fc + 30 MHz mask break point (20 MHz)	Number+	dBr	–100	0	–45	0.01 dB
7	fc – 60 MHz mask break point (40 MHz)	Number+	dBr	–100	0	–45	0.01 dB
8	fc – 40 MHz mask break point (40 MHz)	Number+	dBr	–100	0	–28	0.01 dB
9	fc – 21 MHz mask break point (40 MHz)	Number+	dBr	–100	0	–20	0.01 dB
10	fc + 21 MHz mask break point (40 MHz)	Number+	dBr	–100	0	–20	0.01 dB
11	fc + 40 MHz mask break point (40 MHz)	Number+	dBr	–100	0	–28	0.01 dB
12	fc + 60 MHz mask break point (40 MHz)	Number+	dBr	–100	0	–45	0.01 dB

Example of Use

```
:CONF:SRW:WLN:SMAS:ULIM -45.00,-28.00,-20.00,-20.00,-28.00,-45.00,
                        -45.00,-28.00,-20.00,-20.00,-28.00,-45.00

:CONF:SRW:WLN:SMAS:ULIM?
> -45.00,-28.00,-20.00,-20.00,-28.00,-45.00,
-45.00,-28.00,-20.00,-20.00,-28.00,-45.00
```

:CONFigure:SRWireless:WLN:TCFLeakage:ULIMit

Function

Set or query the WLAN N (OFDM) transmitter center frequency leakage upper limit.

Command

```
:CONFigure:SRWireless:WLN:TCFLeakage:ULIMit
```

Note:

If any parameter is out of range, the command will fail and all the original parameter values will be retained.

Query

```
:CONFigure:SRWireless:WLN:TCFLeakage:ULIMit?
```

Response

#	Description	Format	Units
1	Transmitter center frequency leakage upper limit (20 MHz channel)	Real number	dBr
2	Transmitter center frequency leakage upper limit (40 MHz channel)	Real number	dBr
3	Transmitter center frequency leakage upper limit (upper / lower 20 MHz TX in 40 MHz channel) and 40 MHz duplicate	Real number	dBr

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Transmitter center frequency leakage upper limit (20 MHz channel)	Number+	dB	–100	100	–15	0.01 dB
2	Transmitter center frequency leakage upper limit (40 MHz channel)	Number+	dB	–100	100	–20	0.01 dB
3	Transmitter center frequency leakage upper limit (upper / lower 20 MHz TX in 40 MHz channel) and 40 MHz duplicate	Number+	dB	–100	100	–17	0.01 dB

Example of Use

```
:CONF:SRW:WLN:TCFL:ULIM -15,-20,-17
```

:CONFigure:SRWireless:WLN:TCFTolerance:LIMit

Function

Set or query the WLAN N (OFDM) transmitter center frequency tolerance limit (±).

Command

```
:CONFigure:SRWireless:WLN:TCFTolerance:LIMit
```

Note:

If any parameter is out of range, the command will fail and all the original parameter values will be retained.

Query

```
:CONFigure:SRWireless:WLN:TCFTolerance:LIMit?
```

Response

#	Description	Format	Units
1	Transmitter center frequency tolerance limit (±) 2.4 GHz band	Real number	ppm
2	Transmitter center frequency tolerance limit (±) 5 GHz band	Real number	ppm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Transmitter center frequency tolerance limit (±) 2.4 GHz band	Number+	ppm	0	200	25	0.1 ppm
2	Transmitter center frequency tolerance limit (±) 5 GHz band	Number+	ppm	0	200	20	0.1 ppm

Example of Use

```
:CONF:SRW:WLN:TCFT:LIM 23, 24          // Set to ±23 ppm in 2.4 GHz band
                                       // and ±24 ppm in 5 GHz band.
```

:CONFigure:SRWireless:WLP:DRATe

Function

Set or query the expected data rate and bandwidth of WLAN P packets in this segment.

Command

```
:CONFigure:SRWireless:WLP:DRATe
```

Query

```
:CONFigure:SRWireless:WLP:DRATe?
```

Response

#	Description	Format
1	WLAN P data rate and bandwidth	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	WLAN P data rate and bandwidth	See definition	---	---	---	20MHZ_54MBPS

WLAN P data rate and bandwidth:

5MHZ_1M5BPS	1.5 Mbps (5 MHz BW)
5MHZ_2M25BPS	2.25 Mbps (5 MHz BW)
5MHZ_3MBPS	3 Mbps (5 MHz BW)
5MHZ_4M5BPS	4.5 Mbps (5 MHz BW)
5MHZ_6MBPS	6 Mbps (5 MHz BW)
5MHZ_9MBPS	9 Mbps (5 MHz BW)
5MHZ_12MBPS	12 Mbps (5 MHz BW)
5MHZ_13M5BPS	13.5 Mbps (5 MHz BW)
10MHZ_3MBPS	3 Mbps (10 MHz BW)
10MHZ_4M5BPS	4.5 Mbps (10 MHz BW)
10MHZ_6MBPS	6 Mbps (10 MHz BW)
10MHZ_9MBPS	9 Mbps (10 MHz BW)
10MHZ_12MBPS	12 Mbps (10 MHz BW)
10MHZ_18MBPS	18 Mbps (10 MHz BW)
10MHZ_24MBPS	24 Mbps (10 MHz BW)
10MHZ_27MBPS	27 Mbps (10 MHz BW)
20MHZ_6MBPS	6 Mbps (20 MHz BW)
20MHZ_9MBPS	9 Mbps (20 MHz BW)
20MHZ_12MBPS	12 Mbps (20 MHz BW)
20MHZ_18MBPS	18 Mbps (20 MHz BW)
20MHZ_24MBPS	24 Mbps (20 MHz BW)
20MHZ_36MBPS	36 Mbps (20 MHz BW)

20MHZ_48MBPS	48 Mbps (20 MHz BW)
20MHZ_54MBPS	54 Mbps (20 MHz BW)

Example of Use

```
:CONF:SRW:WLP:DRAT 10MHZ_27MBPS     // Measure a 10 MHz, 27 Mbps signal.
```

:CONFigure:SRWireless:WLP:EVM:ULIMit

Function

Set or query the WLAN P EVM upper limits.

Command

```
:CONFigure:SRWireless:WLP:EVM:ULIMit
```

Note:

If any parameter is out of range, the command will fail and all the original parameter values will be retained.

Query

```
:CONFigure:SRWireless:WLP:EVM:ULIMit?
```

Response

#	Description	Format	Units
1	WLAN P EVM upper limit (5 MHz, 1.5 Mbps)	Real number	dB
2	WLAN P EVM upper limit (5 MHz, 2.25 Mbps)	Real number	dB
3	WLAN P EVM upper limit (5 MHz, 3 Mbps)	Real number	dB
4	WLAN P EVM upper limit (5 MHz, 4.5 Mbps)	Real number	dB
5	WLAN P EVM upper limit (5 MHz, 6 Mbps)	Real number	dB
6	WLAN P EVM upper limit (5 MHz, 9 Mbps)	Real number	dB
7	WLAN P EVM upper limit (5 MHz, 12 Mbps)	Real number	dB
8	WLAN P EVM upper limit (5 MHz, 13.5 Mbps)	Real number	dB
9	WLAN P EVM upper limit (10 MHz, 3 Mbps)	Real number	dB
10	WLAN P EVM upper limit (10 MHz, 4.5 Mbps)	Real number	dB
11	WLAN P EVM upper limit (10 MHz, 6 Mbps)	Real number	dB
12	WLAN P EVM upper limit (10 MHz, 9 Mbps)	Real number	dB
13	WLAN P EVM upper limit (10 MHz, 12 Mbps)	Real number	dB
14	WLAN P EVM upper limit (10 MHz, 18 Mbps)	Real number	dB
15	WLAN P EVM upper limit (10 MHz, 24 Mbps)	Real number	dB
16	WLAN P EVM upper limit (10 MHz, 27 Mbps)	Real number	dB
17	WLAN P EVM upper limit (20 MHz, 6 Mbps)	Real number	dB
18	WLAN P EVM upper limit (20 MHz, 9 Mbps)	Real number	dB
19	WLAN P EVM upper limit (20 MHz, 12 Mbps)	Real number	dB
20	WLAN P EVM upper limit (20 MHz, 18 Mbps)	Real number	dB
21	WLAN P EVM upper limit (20 MHz, 24 Mbps)	Real number	dB
22	WLAN P EVM upper limit (20 MHz, 36 Mbps)	Real number	dB
23	WLAN P EVM upper limit (20 MHz, 48 Mbps)	Real number	dB
24	WLAN P EVM upper limit (20 MHz, 54 Mbps)	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	WLAN P EVM upper limit (5 MHz, 1.5 Mbps)	Number+	dB	–100	100	–5	0.1 dB
2	WLAN P EVM upper limit (5 MHz, 2.25 Mbps)	Number+	dB	–100	100	–8	0.1 dB
3	WLAN P EVM upper limit (5 MHz, 3 Mbps)	Number+	dB	–100	100	–10	0.1 dB
4	WLAN P EVM upper limit (5 MHz, 4.5 Mbps)	Number+	dB	–100	100	–13	0.1 dB
5	WLAN P EVM upper limit (5 MHz, 6 Mbps)	Number+	dB	–100	100	–16	0.1 dB
6	WLAN P EVM upper limit (5 MHz, 9 Mbps)	Number+	dB	–100	100	–19	0.1 dB
7	WLAN P EVM upper limit (5 MHz, 12 Mbps)	Number+	dB	–100	100	–22	0.1 dB
8	WLAN P EVM upper limit (5 MHz, 13.5 Mbps)	Number+	dB	–100	100	–25	0.1 dB
9	WLAN P EVM upper limit (10 MHz, 3 Mbps)	Number+	dB	–100	100	–5	0.1 dB
10	WLAN P EVM upper limit (10 MHz, 4.5 Mbps)	Number+	dB	–100	100	–8	0.1 dB
11	WLAN P EVM upper limit (10 MHz, 6 Mbps)	Number+	dB	–100	100	–10	0.1 dB
12	WLAN P EVM upper limit (10 MHz, 9 Mbps)	Number+	dB	–100	100	–13	0.1 dB
13	WLAN P EVM upper limit (10 MHz, 12 Mbps)	Number+	dB	–100	100	–16	0.1 dB
14	WLAN P EVM upper limit (10 MHz, 18 Mbps)	Number+	dB	–100	100	–19	0.1 dB
15	WLAN P EVM upper limit (10 MHz, 24 Mbps)	Number+	dB	–100	100	–22	0.1 dB
16	WLAN P EVM upper limit (10 MHz, 27 Mbps)	Number+	dB	–100	100	–25	0.1 dB
17	WLAN P EVM upper limit (20 MHz, 6 Mbps)	Number+	dB	–100	100	–5	0.1 dB
18	WLAN P EVM upper limit (20 MHz, 9 Mbps)	Number+	dB	–100	100	–8	0.1 dB
19	WLAN P EVM upper limit (20 MHz, 12 Mbps)	Number+	dB	–100	100	–10	0.1 dB
20	WLAN P EVM upper limit (20 MHz, 18 Mbps)	Number+	dB	–100	100	–13	0.1 dB
21	WLAN P EVM upper limit (20 MHz, 24 Mbps)	Number+	dB	–100	100	–16	0.1 dB

22	WLAN P EVM upper limit (20 MHz, 36 Mbps)	Number+	dB	–100	100	–19	0.1 dB
23	WLAN P EVM upper limit (20 MHz, 48 Mbps)	Number+	dB	–100	100	–22	0.1 dB
24	WLAN P EVM upper limit (20 MHz, 54 Mbps)	Number+	dB	–100	100	–25	0.1 dB

Example of Use

```
:CONF:SRW:WLP:EVM:ULIM -5,-8,-10,-13,-16,-19,-22,-25,
                       -5,-8,-10,-13,-16,-19,-22,-25,
                       -5,-8,-10,-13,-16,-19,-22,-25
```

:CONFigure:SRWireless:WLP:POWer:LLIMit

Function

Define the lower power limit for each IEEE 802.11p power class.

Note:

To use these limits for 802.11p power measurements, set the required power class using `:CONFigure:SRWireless:WLP:POWer:PCLass`.

Command

```
:CONFigure:SRWireless:WLP:POWer:LLIMit
```

Query

```
:CONFigure:SRWireless:WLP:POWer:LLIMit?
```

Response

#	Description	Format	Units	Resolution
1	Transmitter average power lower limit (power class A)	Real number	dBm	0.01 dB
2	Transmitter average power lower limit (power class B)	Real number	dBm	0.01 dB
3	Transmitter average power lower limit (power class C)	Real number	dBm	0.01 dB
4	Transmitter average power lower limit (power class D)	Real number	dBm	0.01 dB

Parameters

#	Description	Format	Units	Min	Max	Def	Resolution
1	Transmitter average power lower limit (power class A)	Number+	dBm	–100	100	–40	0.01 dB
2	Transmitter average power lower limit (power class B)	Number+	dBm	–100	100	–30	0.01 dB
3	Transmitter average power lower limit (power class C)	Number+	dBm	–100	100	–20	0.01 dB
4	Transmitter average power lower limit (power class D)	Number+	dBm	–100	100	–11.19	0.01 dB

Example of Use

```
:CONF:SRW:WLP:POW:LLIM -40,7.5,-20.5,-11.19       // Set the limits
:CONF:SRW:WLP:POW:PCL B                           // Set Power Class B
```

:CONFigure:SRWireless:WLP:POWer:PCLass

Function

Set the IEEE 802.11p power class for transmit power.
See also `:CONFigure:SRWireless:WLP:POWer:ULIMit`
and `:CONFigure:SRWireless:WLP:POWer:LLIMit`.

The following table shows the relationship between the power classes and the upper and lower transmitted power limits. When the power class is "OFF" the limits used are same as for all other WLAN standards.

POWER CLASS	802.11p Lower Limit (dBm)	802.11p Upper Limit (dBm)
OFF	Value set using: `:CONF:SRW:POW:LLIM`	Value set using: `:CONF:SRW:POW:ULIM`
A	–40.00	0.00
B	–30.00	10.00
C	–20.00	20.00
D	–11.19	28.81

Command

`:CONFigure:SRWireless:WLP:POWer:PCLass`

Query

`:CONFigure:SRWireless:WLP:POWer:PCLass?`

Response

#	Description	Format
1	Power class	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	Power class	See definition	---	---	---	OFF

Power class:

OFF — Power class limits are not used. The limits common to all wireless standards are applied. These can be changed using: `:CONF:SRW:WLP:POWer:ULIM` and `:CONF:WLP:POW:LLIM`.

A to D — Apply the corresponding power class limits.

Example of Use

```
:CONF:SRW:WLP:POW:PCL B             // Set Power Class B
```

:CONFigure:SRWireless:WLP:POWer:ULIMit

Function

Define the upper power limit for each IEEE 802.11p power class.

Note:

To use these limits for 802.11p power measurements, set the required power class using `:CONFigure:SRWireless:WLP:POWer:PCLass`.

Command

```
:CONFigure:SRWireless:WLP:POWer:ULIMit
```

Query

```
:CONFigure:SRWireless:WLP:POWer:ULIMit?
```

Response

#	Description	Format	Units	Resolution
1	Transmitter average power upper limit (power class A)	Real number	dBm	0.01 dB
2	Transmitter average power upper limit (power class B)	Real number	dBm	0.01 dB
3	Transmitter average power upper limit (power class C)	Real number	dBm	0.01 dB
4	Transmitter average power upper limit (power class D)	Real number	dBm	0.01 dB

Parameters

#	Description	Format	Units	Min	Max	Def	Resolution
1	Transmitter average power upper limit (power class A)	Number+	dBm	–100	100	0	0.01 dB
2	Transmitter average power upper limit (power class B)	Number+	dBm	–100	100	10	0.01 dB
3	Transmitter average power upper limit (power class C)	Number+	dBm	–100	100	20	0.01 dB
4	Transmitter average power upper limit (power class D)	Number+	dBm	–100	100	28.81	0.01 dB

Example of Use

```
:CONF:SRW:WLP:POW:ULIM 0,9.5,20,28.81      // Set the limits
:CONF:SRW:WLP:POW:PCL B                   // Set Power Class B
```

:CONFigure:SRWireless:WLP:SCFTolerance:LIMit

Function

Set or query the WLAN P (OFDM) symbol clock frequency tolerance limit (±).

Command

```
:CONFigure:SRWireless:WLP:SCFTolerance:LIMit
```

Query

```
:CONFigure:SRWireless:WLP:SCFTolerance:LIMit?
```

Response

#	Description	Format	Units
1	Symbol clock frequency tolerance limit (±) (5 MHz bandwidth)	Real number	ppm
2	Symbol clock frequency tolerance limit (±) (10 MHz bandwidth)	Real number	ppm
3	Symbol clock frequency tolerance limit (±) (20 MHz bandwidth)	Real number	ppm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Symbol clock frequency tolerance limit (±) (5 MHz bandwidth)	Number+	ppm	0	200	10	0.1 ppm
2	Symbol clock frequency tolerance limit (±) (10 MHz bandwidth)	Number+	ppm	0	200	20	0.1 ppm
3	Symbol clock frequency tolerance limit (±) (20 MHz bandwidth)	Number+	ppm	0	200	20	0.1 ppm

Example of Use

```
:CONF:SRW:WLP:SCFT:LIM 10,20,20     // Set to ±10 ppm for 5 MHz signals,
                                              // and ±20 ppm for 10 and
20 MHz signals
```

:CONFigure:SRWireless:WLP:SFLatness:LIMit

Function

Set or query the WLAN P (OFDM) spectral flatness limits.

Command

```
:CONFigure:SRWireless:WLP:SFLatness:LIMit
```

Note:

If any parameter is out of range, the command will fail and all the original parameter values will be retained.

Query

```
:CONFigure:SRWireless:WLP:SFLatness:LIMit?
```

Response

#	Description	Format	Units
1	Upper limit: subcarriers –26 to –17	Real number	dB
2	Lower limit: subcarriers –26 to –17	Real number	dB
3	Upper limit: subcarriers –16 to –1	Real number	dB
4	Lower limit: subcarriers –16 to –1	Real number	dB
5	Upper limit: subcarriers +1 to +16	Real number	dB
6	Lower limit: subcarriers +1 to +16	Real number	dB
7	Upper limit: subcarriers +17 to +26	Real number	dB
8	Lower limit: subcarriers +17 to +26	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Upper limit: subcarriers –26 to –17	Number+	dB	–100	100	4	0.01 dB
2	Lower limit: subcarriers –26 to –17	Number+	dB	–100	100	–6	0.01 dB
3	Upper limit: subcarriers –16 to –1	Number+	dB	–100	100	4	0.01 dB
4	Lower limit: subcarriers -16 to –1	Number+	dB	–100	100	–4	0.01 dB
5	Upper limit: subcarriers +1 to +16	Number+	dB	–100	100	4	0.01 dB
6	Lower limit: subcarriers +1 to +16	Number+	dB	–100	100	–4	0.01 dB
7	Upper limit: subcarriers +17 to +26	Number+	dB	–100	100	4	0.01 dB
8	Lower limit: subcarriers +17 to +26	Number+	dB	–100	100	–6	0.01 dB

Example of Use

```
:CONF:SRW:WLP:SFL:LIM 4,-6,4,-4,4,-4,4,-6
```

:CONFigure:SRWireless:WLP:SMASk:PCLass

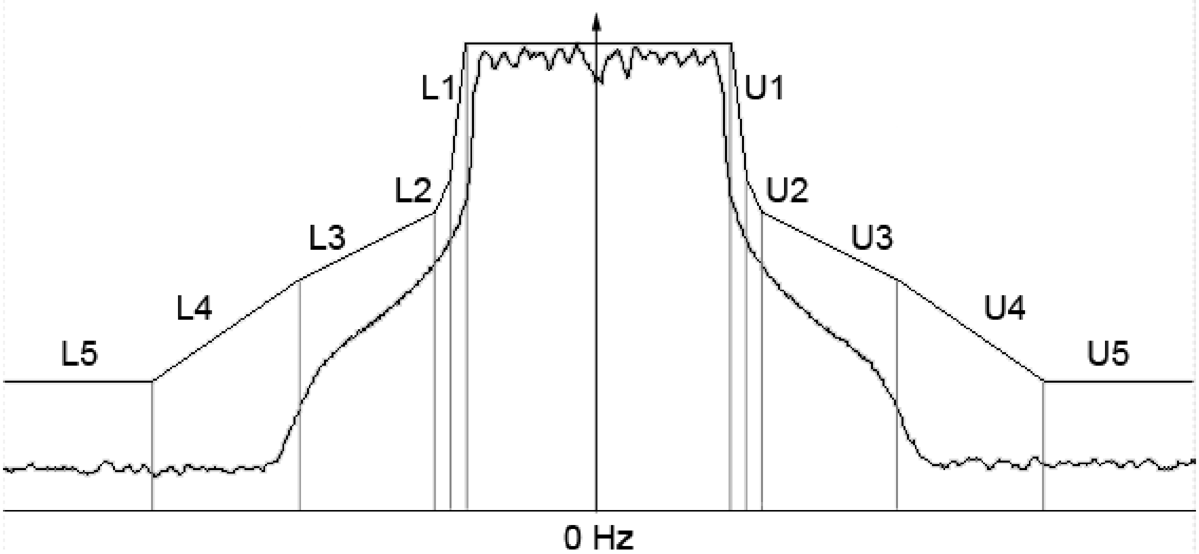
Function

Set the IEEE 802.11p power class for spectral mask measurements.
See also `:CONFigure:SRWireless:WLP:SMASk:PCLass:ULIMit`.

The following table shows the relationship between the power classes and spectral mask limits. When the power class is “OFF” the mask defined by `:CONF:SRW:WLP:SMAS:ULIM` is used.

Bandwidth (MHz)	5	10	20	5	10	20	5	10	20	5	10	20
Frequency offset from carrier (MHz)	±2.5	±5	±10	±2.75	±5.5	±11	±5	±10	±20	±7.5	±15	±30
Power Class A	–10 dBr			–20 dBr			–28 dBr			–40 dBr		
Power Class B	–16 dBr			–20 dBr			–28 dBr			–40 dBr		
Power Class C	–26 dBr			–32 dBr			–40 dBr			–50 dBr		
Power Class D	–35 dBr			–45 dBr			–55 dBr			–65 dBr		

Between the carrier frequency and ±2.25 MHz, ±4.5 MHz or ±9 MHz (for 5 MHz, 10 MHz and 20 MHz bandwidth respectively) the mask value is 0 dBr by definition. See figure below.



	Frequency offset from carrier (MHz)									
	L5	L4	L3	L2	L1	U1	U2	U3	U4	U5
5 MHz	–7.5	–5	–2.75	–2.5	–2.25	+2.25	+2.5	+2.75	+5	+7.5
10 MHz	–15	–10	–5.5	–5	–4.5	+4.5	+5	+5.5	+10	+15
20 MHz	–30	–20	–11	–10	–9	+9	+10	+11	+20	+30

The table shows the frequency offsets from the carrier of the start of each segment of the spectral mask for each signal bandwidth. Note that there are 10 segments (L5 to U5) compared with the IEEE standard mask, which has 8 segments (L4 to U4).

Command

```
:CONFigure:SRWireless:WLP:POWer:PCLass
```

Query

```
:CONFigure:SRWireless:WLP:POWer:PCLass?
```

Response

#	Description	Format
1	Power class	See definition

Parameters

#	Description	Format	Units	Min	Max	Default
1	Power class	See definition	---	---	---	OFF

Power class:

OFF	A power class spectral mask is not used. Instead the mask defined by `:CONF:SRW:WLP:SMAS:ULIM` is used.
A to D	Apply the corresponding power class spectral mask.

Example of Use

```
:CONF:SRW:WLP:SMAS:PCL B          // Use the spectral mask for Power
                                  // Class B
```

:CONFigure:SRWireless:WLP:SMASk:PCLass:ULIMit

Function

Define the spectral mask for an IEEE 802.11p power class.

Note:
To use these limits for 802.11p spectral profile measurements, set the required power class using `:CONFigure:SRWireless:WLP:SMASk:PCLass`.

Command

`:CONFigure:SRWireless:WLP:SMASk:PCLass:ULIMit`

Query

`:CONFigure:SRWireless:WLP:SMASk:PCLass:ULIMit?`

Response

#	Description	Format	Units	Resolution
1	Power Class (possible values are A, B, C or D)	Character	---	---
2	Bandwidth (possible values are 5, 10 or 20	Integer	MHz	1 MHz
3	Mask value at ± 2.5 MHz (5 MHz BW), ± 5.0 MHz (10 MHz BW), ± 10 MHz (20 MHz BW)	Real number	dBr	0.01 dB
4	Mask value at ± 2.75 MHz (5 MHz BW), ± 5.5 MHz (10 MHz BW) or ± 11 MHz (20 MHz BW)	Real number	dBr	0.01 dB
5	Mask value at ± 5.0 MHz (5 MHz BW), ± 10.0 MHz (10 MHz BW) or ± 20.0 MHz (20 MHz BW)	Real number	dBr	0.01 dB
6	Mask value at ± 7.5 MHz (5 MHz BW), ± 15.0 MHz (10 MHz BW) or ± 30 MHz (20 MHz BW)	Real number	dBr	0.01 dB

Parameters

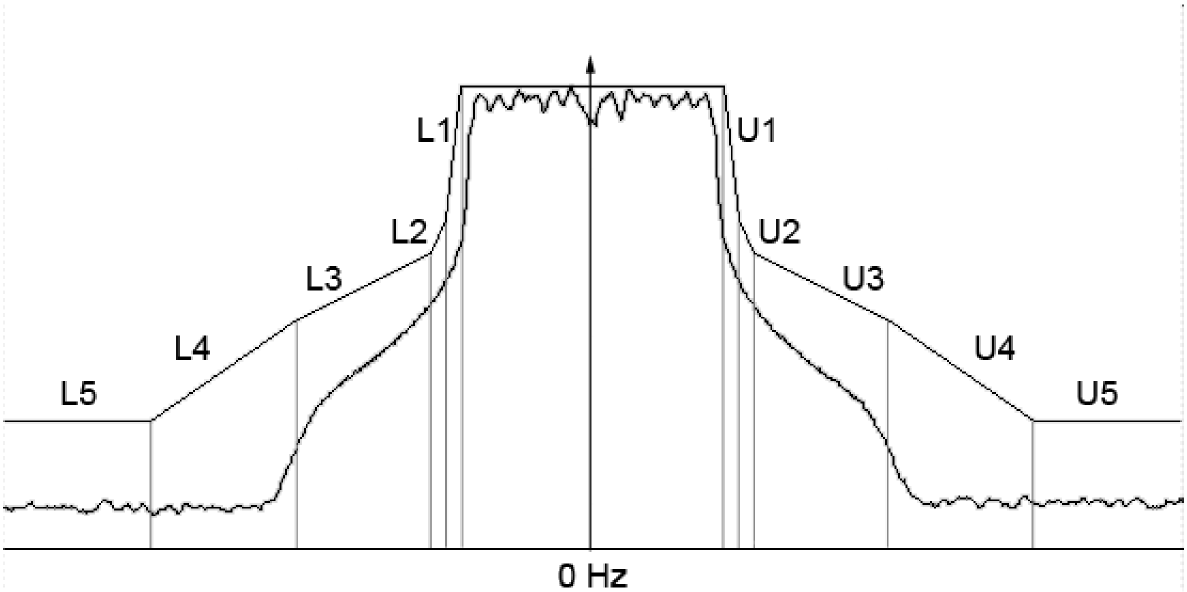
#	Description	Format	Units	Min	Max	Def	Res
1	Power Class (possible values are A, B, C or D)	Character	---	---	---	---	---
2	Bandwidth (possible values are 5, 10 or 20	Number+	dBr	–100	0	See table	0.01 dB
3	Mask value at ± 2.5 MHz (5 MHz BW), ± 5.0 MHz (10 MHz BW), ± 10 MHz (20 MHz BW)	Number+	dBr	–100	0	See table	0.01 dB
4	Mask value at ± 2.75 MHz (5 MHz BW), ± 5.5 MHz (10 MHz BW) or ± 11 MHz (20 MHz BW)	Number+	dBr	–100	0	See table	0.01 dB
5	Mask value at ± 5.0 MHz (5 MHz BW), ± 10.0 MHz (10 MHz BW) or ± 20.0 MHz (20 MHz BW)	Number+	dBr	–100	0	See table	0.01 dB

6	Mask value at ± 7.5 MHz (5 MHz BW), ± 15.0 MHz (10 MHz BW) or ± 30 MHz (20 MHz BW)	Number+	dBr	–100	0	See table	0.01 dB

Default Values

Bandwidth (MHz)	5	10	20	5	10	20	5	10	20	5	10	20
Frequency offset from carrier (MHz)	±2.5	±5	±10	±2.75	±5.5	±11	±5	±10	±20	±7.5	±15	±30
Power Class A	–10 dBr			–20 dBr			–28 dBr			–40 dBr		
Power Class B	–16 dBr			–20 dBr			–28 dBr			–40 dBr		
Power Class C	–26 dBr			–32 dBr			–40 dBr			–50 dBr		
Power Class D	–35 dBr			–45 dBr			–55 dBr			–65 dBr		

Between the carrier frequency and offsets of ±2.25 MHz, ±4.5 MHz or ±9 MHz (for 5 MHz, 10 MHz and 20 MHz bandwidth respectively) the mask value is 0 dBr by definition. See figure below:



	Frequency offset from carrier (MHz)									
	L5	**L4**	**L3**	**L2**	**L1**	**U1**	**U2**	**U3**	**U4**	**U5**
5 MHz	–7.5	–5	–2.75	–2.5	–2.25	+2.25	+2.5	+2.75	+5	+7.5
10 MHz	–15	–10	–5.5	–5	–4.5	+4.5	+5	+5.5	+10	+15
20 MHz	–30	–20	–11	–10	–9	+9	+10	+11	+20	+30

The table shows the frequency offsets from the carrier of the start of each segment of the spectral mask. Note that there are 10 segments (labelled L5 to U5) compared with the IEEE standard mask, which has 8 segments (labelled L4 to U4).

Example of Use

```
:CONF:SRW:WLP:SMAS:PCL:ULIM B,5,-13,-17,-25,-37   // Adjust the limits
                                                  // for Power Class B 5 MHz

:CONF:SRW:WLP:SMAS:PCL B                          // Set Power Class B
```

:CONFigure:SRWireless:WLP:SMASk:ULIMit

Function

Define the spectral mask limits for WLAN IEEE 802.11p signals.

Command

```
:CONFigure:SRWireless:WLP:SMASk:ULIMit
```

Query

```
:CONFigure:SRWireless:WLP:SMASk:ULIMit?
```

Response

#	Description	Format	Units	Resolution
1	fc – 7.5 MHz mask break point (5 MHz BW)	Real number	dBr	0.01 dB
2	fc – 5 MHz mask break point (5 MHz BW)	Real number	dBr	0.01 dB
3	fc – 2.75 MHz mask break point (5 MHz BW)	Real number	dBr	0.01 dB
4	fc + 2.75 MHz mask break point (5 MHz BW)	Real number	dBr	0.01 dB
5	fc + 5 MHz mask break point (5 MHz BW)	Real number	dBr	0.01 dB
6	fc + 7.5 MHz mask break point (5 MHz BW)	Real number	dBr	0.01 dB
7	fc – 15 MHz mask break point (10 MHz BW)	Real number	dBr	0.01 dB
8	fc – 10 MHz mask break point (10 MHz BW)	Real number	dBr	0.01 dB
9	fc – 5.5 MHz mask break point (10 MHz BW)	Real number	dBr	0.01 dB
10	fc + 5.5 MHz mask break point (10 MHz BW)	Real number	dBr	0.01 dB
11	fc + 10 MHz mask break point (10 MHz BW)	Real number	dBr	0.01 dB
12	fc + 15 MHz mask break point (10 MHz BW)	Real number	dBr	0.01 dB
13	fc – 30 MHz mask break point (20 MHz BW)	Real number	dBr	0.01 dB
14	fc – 20 MHz mask break point (20 MHz BW)	Real number	dBr	0.01 dB
15	fc – 11 MHz mask break point (20 MHz BW)	Real number	dBr	0.01 dB
16	fc + 11 MHz mask break point (20 MHz BW)	Real number	dBr	0.01 dB
17	fc + 20 MHz mask break point (20 MHz BW)	Real number	dBr	0.01 dB
18	fc + 30 MHz mask break point (20 MHz BW)	Real number	dBr	0.01 dB

Parameters

#	Description	Format	Units	Min	Max	Def	Resolution
1	fc – 7.5 MHz mask break point (5 MHz BW)	Number+	dBr	–100	0	–40	0.01 dB
2	fc – 5 MHz mask break point (5 MHz BW)	Number+	dBr	–100	0	–28	0.01 dB
3	fc – 2.75 MHz mask break point (5 MHz BW)	Number+	dBr	–100	0	–20	0.01 dB
4	fc + 2.75 MHz mask break point (5 MHz BW)	Number+	dBr	–100	0	–20	0.01 dB
5	fc + 5 MHz mask break point	Number+	dBr	–100	0	–28	0.01 dB

	(5 MHz BW)						
6	fc + 7.5 MHz mask break point (5 MHz BW)	Number+	dBr	–100	0	–40	0.01 dB
7	fc – 15 MHz mask break point (10 MHz BW)	Number+	dBr	–100	0	–40	0.01 dB
8	fc – 10 MHz mask break point (10 MHz BW)	Number+	dBr	–100	0	–28	0.01 dB
9	fc – 5.5 MHz mask break point (10 MHz BW)	Number+	dBr	–100	0	–20	0.01 dB
10	fc + 5.5 MHz mask break point (10 MHz BW)	Number+	dBr	–100	0	–20	0.01 dB
11	fc + 10 MHz mask break point (10 MHz BW)	Number+	dBr	–100	0	–28	0.01 dB
12	fc + 15 MHz mask break point (10 MHz BW)	Number+	dBr	–100	0	–40	0.01 dB
13	fc – 30 MHz mask break point (20 MHz BW)	Number+	dBr	–100	0	–40	0.01 dB
14	fc – 20 MHz mask break point (20 MHz BW)	Number+	dBr	–100	0	–28	0.01 dB
15	fc – 11 MHz mask break point (20 MHz BW)	Number+	dBr	–100	0	–20	0.01 dB
16	fc + 11 MHz mask break point (20 MHz BW)	Number+	dBr	–100	0	–20	0.01 dB
17	fc + 20 MHz mask break point (20 MHz BW)	Number+	dBr	–100	0	–28	0.01 dB
18	fc + 30 MHz mask break point (20 MHz BW)	Number+	dBr	–100	0	–40	0.01 dB

Example of Use

```
:CONF:SRW:WLP:SMAS:ULIM -40.00,-28.00,-20.00,-20.00,-28.00,-40.00,
                        -40.00,-28.00,-20.00,-20.00,-28.00,-40.00,
                        -40.00,-28.00,-20.00,-20.00,-28.00,-40.00
```

:CONFigure:SRWireless:WLP:TCFLeakage:ULIMit

Function

Set or query the WLAN P (OFDM) transmitter center frequency leakage upper limit.

Command

```
:CONFigure:SRWireless:WLP:TCFLeakage:ULIMit
```

Query

```
:CONFigure:SRWireless:WLP:TCFLeakage:ULIMit?
```

Response

#	Description	Format	Units
1	Transmitter center frequency leakage upper limit	Real number	dB

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Transmitter center frequency leakage upper limit	Number+	dB	–100	100	–15	0.01 dB

Example of Use

```
:CONF:SRW:WLP:TCFL:ULIM -14
```

:CONFigure:SRWireless:WLP:TCFTolerance:LIMit

Function

Set or query the WLAN P (OFDM) transmitter center frequency tolerance limit (±).

Command

```
:CONFigure:SRWireless:WLP:TCFTolerance:LIMit
```

Query

```
:CONFigure:SRWireless:WLP:TCFTolerance:LIMit?
```

Response

#	Description	Format	Units
1	Transmitter center frequency tolerance limit (±) (5 MHz bandwidth)	Real number	ppm
2	Transmitter center frequency tolerance limit (±) (10 MHz bandwidth)	Real number	ppm
3	Transmitter center frequency tolerance limit (±) (20 MHz bandwidth)	Real number	ppm

Parameters

#	Description	Format	Units	Min	Max	Default	Resolution
1	Transmitter center frequency tolerance limit (±) (5 MHz bandwidth)	Number+	ppm	0	200	10	0.1 ppm
2	Transmitter center frequency tolerance limit (±) (10 MHz bandwidth)	Number+	ppm	0	200	20	0.1 ppm
3	Transmitter center frequency tolerance limit (±) (20 MHz bandwidth)	Number+	ppm	0	200	20	0.1 ppm

Example of Use

```
:CONF:SRW:WLP:TCFT:LIM 10,20,20     // Set to ±10 ppm for 5 MHz signals
                                    // and ±20 ppm for 10 and 20 MHz signals.
```

:FETCh:SRWireless:CINFormation?

Function

Fetch capture information (segment locations, packet edge locations and measurement status).

Command

N/A

Query

```
:FETCh:SRWireless:CINFormation?
```

Response

#	Description	Format
1	Capture status	Integer
2	Number of segments	Integer
	For each segment:	
1	Segment index	Integer
2	Segment status	Integer
3	Offset from start of capture in IQ sample pairs	Integer
4	Width of segment in IQ sample pairs	Integer
5	Number of packets in segment	Integer
	For each packet	
1	Packet index	Integer
2	Packet status	Integer
3	Offset from start of segment in IQ sample pairs	Integer
4	Width of packet in IQ sample pairs	Integer

Parameters

N/A

Example of Use

```
:FETC:SRW:CINF?

> 0,1,1,0,0,200000,2,1,0,1000,80000,2,0,100000,80000
```

Field	Description
0	Capture status: No problem found
1	Number of segments: 1
1	First segment index: 1
0	First segment status: No problem found

0	First segment offset from start of capture: 0 samples.
200000	First segment width: 200,000 samples
2	First segment > Number of packets: 2
1	First segment, first packet index: 1
0	First segment, first packet status: No problem found
1000	First segment, first packet offset from start of segment: 1000 samples
80000	First segment, first packet width: 80000 samples
2	First segment, second packet index: 2
0	First segment, second packet status: No problem found
100000	First segment, second packet offset from start of segment: 100000 samples
80000	First segment, second packet width: 80000 samples

:FETCh:SRWireless:CINFormation? (Extended)

Function

Fetch capture information (segment locations, packet edge locations and measurement status). An optional extension code returns additional information as described in the following table:

Extension code bit	Additional fields in response
0	Sampling rate used by VSA for each segment

Command

N/A

Query

```
:FETCh:SRWireless:CINFormation?
```

Response

#	Description	Format
1	Capture status	Integer
2	Extension code	Integer
3	Number of segments	Integer
	For each segment:	
1	Segment index	Integer
2	Segment status	Integer
3	Offset from start of capture in IQ sample pairs	Integer
4	Width of segment in IQ sample pairs	Integer
5	Number of packets in segment	Integer
	If bit 0 of extension code set:	
6	Sampling rate (Hz)	Integer
	For each packet:	
1	Packet index	Integer
2	Packet status	Integer
3	Offset from start of segment in IQ sample pairs	Integer
4	Width of packet in IQ sample pairs	Integer

Parameters

#	Description	Format
1	Extension code	Number

Example of Use

```
:FETC:SRW:CINF? 1                    // Request capture information
                                     // using an extension code with
                                     // bit 0 set.

> 0,1,1,1,0,0,200000,2,200000000,1,0,1000,80000,2,0,100000,80000
```

Field	Description
0	Capture status: No problem found
1	Extension code: bit 0 is set
1	Number of segments: 1
1	First segment index: 1
0	First segment status: No problem found
0	First segment offset from start of capture: 0 samples.
200000	First segment width: 200,000 samples
2	First segment > Number of packets: 2
200000000	First segment sampling rate: 200 MS/s
1	First segment, first packet index: 1
0	First segment, first packet status: No problem found
1000	First segment, first packet offset from start of segment: 1000 samples
80000	First segment, first packet width: 80000 samples
2	First segment, second packet index: 2
0	First segment, second packet status: No problem found
100000	First segment, second packet offset from start of segment: 100000 samples
80000	First segment, second packet width: 80000 samples

:FETCh:SRWireless:CPRofile?

Function

Fetch profile data from a specified portion of the capture - binary format.

Command

N/A

Query

`:FETCh:SRWireless:CPRofile?`

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Number of points (n)	INT32	---
2+1	First power reading	REAL32	dBm
…	…		
2+n	Last power reading	REAL32	dBm

Parameters

#	Description	Format	Units	Min	Max	Default
1	Start position in IQ sample pairs	Number	---	0	See notes.	---
2	Number of points	Number	---	1	See notes	---

Notes:

1. Use `:FETCh:SRW:CINFormation?` to determine the positions of areas of interest within the capture buffer. For example, the position and width of each segment within the capture and the locations of packets within each segment. This information may then be used to select a suitable start position and number of points for `:FETCh:SRWireless:CPRofile?`
2. It is not possible to extract data outside the range of the most recent capture.
3. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Example of Use

```
:FETC:SRW:CPR? 123456, 789123       // Extract log power measurements
                                    // Starting 123,456 samples into the
                                    // capture buffer.  The data length
                                    // is 789123 power readings.
```

:FETCh:SRWireless:IQ?

Function

Fetch IQ data from a specified portion of the capture. - binary format.

Command

N/A

Query

`:FETCh:SRWireless:IQ?`

Response

#	Description	Format	Units
1	#nmmm…mm	Binary header	---
2	Number of IQ Pairs (n)	INT32	---
3	First I value	REAL32	---
4	First Q value	REAL32	---
…	…	…	---
	Last I value	REAL32	---
	Last Q value	REAL32	---

Parameters

#	Description	Format	Units	Min	Max	Default
1	Start position in IQ sample pairs	Number	---	0	See notes.	---
2	Number of points	Number	---	1	See notes	---

Notes:

1. Use `:FETCh:SRW:CINFormation?` to determine the positions of areas of interest within the capture buffer. For example, the position and width of each segment within the capture and the locations of packets within each segment. This information may then be used to select a suitable start position and number of points for `:FETCh:SRWireless:IQ?`
2. It is not possible to extract data outside the range of the most recent capture.
3. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Example of Use

```
:FETC:SRW:IQ? 0,100000      // Extract the first 100000
                            // IQ samples from the start
                            // of the capture.
```

:FETCh:SRWireless:IQ:INFOrmation?

Function

Fetch full capture information, including autolevel and full span spectrum segments, which are not returned by :FETCh:SRWireless:CINFormation?

Notes:

The segment index values returned correspond to those returned by ":FETCh:SRWireless:CINFormation?". In addition to these, the command reports "hidden" segments - for example additional segments introduced by the system to capture full span spectrum measurements. These have high index numbers out of sequence with those of the "visible" segments.

The extension code adds additional results to the response as described in the following table:

Extension code bit	Additional fields in response
0	Segment width in IQ sample pairs (including settling time) Segment offset from start of capture to usable data in IQ sample pairs. (Skips settling time.)
1	Sampling rate in Hz.

Command

N/A

Query

```
:FETCh:SRWireless:IQ:INF?
```

Response

#	Description	Format	Units	Res
1	Extension code	Integer	---	---
2	Number of segments	Integer	---	---
	For each segment:	Integer	---	---
1	Segment index	Integer	---	---
2	Segment status	Integer	---	---
3	Segment offset from start of capture in IQ sample pairs	Integer	---	---
4	Segment width in IQ sample pairs (including settling time).	Integer	---	---
	If bit 0 of extension code set:			
5	Segment offset from start of capture to usable data in IQ sample pairs. (Skips settling time.)	Integer	---	---
6	Segment width in IQ sample pairs (excluding settling time).	Integer	---	---
	If bit 1 of extension code set			
7	Sampling rate	Integer	Hz	1 Hz

Parameters

#	Description	Format
1	Extension code	Number

Example of Use

```
:FETC:SRW:IQ:INFO? 0
```

:FETCh:SRWireless:PACKet:BLE:CDRift?

Function

Fetch *Bluetooth* LE measurements - carrier frequency offset and drift (TRM-LE/CA/BV-06-C, TRM-LE/CA/BV-12-C and TRM-LE/CA/BV-14-C).

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BLE:CDRift?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
4	Maximum frequency offset f_n	Real number	Hz	1 Hz
5	Maximum frequency drift $\|f_0 - f_n\|$	Real number	Hz	1 Hz
6	Maximum drift rate $\|f_n - f_{n-5}\|$	Real number	Hz	1 Hz
7	Maximum frequency offset limit test passed	Boolean	---	---
8	Maximum frequency drift limit test passed	Boolean	---	---
9	Maximum drift rate test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BLE:CDR? 1,2
```

5

Remote Operation

:FETCh:SRWireless:PACKet:BLE:CDRift? (Extended)

Function

Fetch *Bluetooth* LE measurements - carrier frequency offset and drift (TRM-LE/CA/BV-06-C, TRM-LE/CA/BV-12-C and TRM-LE/CA/BV-14-C). An optional extension code returns measurements in addition to those required by the standard.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BLE:CDRift?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code	Integer	---	---
4	Maximum frequency offset f_n	Real number	Hz	1 Hz
5	Maximum frequency drift $\lvert f_0 - f_n \rvert$	Real number	Hz	1 Hz
6	Maximum drift rate $\lvert f_n - f_{n-5} \rvert$	Real number	Hz	1 Hz
7	Maximum frequency offset limit test passed	Boolean	---	---
8	Maximum frequency drift limit test passed	Boolean	---	---
9	Maximum drift rate test passed	Boolean	---	---
	If bit 0 of extension code set:			
10	Initial frequency offset f_0	Real number	Hz	1 Hz
11	Initial frequency drift rate $\lvert f_1 - f_0 \rvert$	Real number	Hz	1 Hz
12	Maximum positive frequency offset f_n	Real number	Hz	1 Hz
13	Maximum negative frequency offset f_n	Real number	Hz	1 Hz
14	Average frequency offset (over all f_n)	Real number	Hz	1 Hz
15	Initial frequency offset limit test passed	Boolean	---	---
16	Initial frequency drift rate limit test passed	Boolean	---	---

Definitions from BLE Standard:

f_0	Carrier frequency offset measured over preamble.
f_1	Carrier frequency offset measured over first 10-bit block of payload.
f_n	Carrier frequency offset measured over the n^{th} 10-bit block of payload.
$\lvert f_n – f_{n-5} \rvert$	Drift rate. This is the absolute value of the difference in carrier frequency offset between a 10-bit block in the payload and that measured five blocks (50 µs) earlier in the payload. In other words, this is the drift rate over a 50 µs time period within the payload. The instrument measures the worst case drift rate over the payload.
$\lvert f_1 – f_0 \rvert$	Initial carrier drift. The absolute value of the difference in carrier frequency offset between the first 10-bit block in the payload and that of the preamble.

$|f_0 - f_n|$ Carrier drift. The absolute value of the difference in carrier frequency offset between any 10-bit block in the payload and that of the preamble. The instrument measures the average and worst case drift across the packet payload.

Note on limit checking extended measurements:

The initial frequency offset limit is 150 kHz by default and may be changed using `:CONF:SRW:BLE:CFOF:LIM`.
The initial frequency drift rate limit is 50 kHz by default and may be changed using `:CONF:SRW:BLE:DRAT:LIM`.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number

Example of Use

```
:FETC:SRW:PACK:BLE:CDR? 1,2,1
```

:FETCh:SRWireless:PACKet:BLE:DEViation:ASCii?

Function

Fetch *Bluetooth* measurements - deviation vs symbol (BLE) – ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BLE:DEViation:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	
2	Packet index	Integer	---	
3	Start time	Real number	s	1 μs
4	Stop time	Real number	s	1 μs
5	Number of points (n)	Integer	---	
6	First deviation point	Real number	Hz	1 Hz
---	…			
	Last deviation point	Real number	Hz	1 Hz
	Index fields valid	Boolean	---	---
	Start offset	Integer	---	---
	Stop offset	Integer	---	---
	Preamble index (p0)	Integer	---	---
	Sync word index	Integer	---	---
	PDU header index	Integer	---	---
	PDU length index	Integer	---	---
	PDU payload index	Integer	---	---
	CRC start index	Integer	---	---
	CRC stop index	Integer	---	---

Notes:

1. The index fields contain offsets from the start of the deviation v symbol array that identify the positions of specific fields. A typical use is for plotting deviation v symbol with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BLE:DEV:ASC? 1,3
```

:FETCh:SRWireless:PACKet:BLE:DEViation[:BINary]?

Function

Fetch *Bluetooth* measurements - deviation vs symbol (BLE) – binary format

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BLE:DEViation[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Start time (relative to p0)	REAL32	s
5	Stop time (relative to p0)	REAL32	s
6	Number of points (n)	INT32	---
7	Position of p0 relative to start of power data	INT32	---
	Deviation in Hz	REAL32	Hz
…	…		
	Index fields valid if non zero.	INT32	---
	Start offset (for use when zooming)	INT32	---
	Stop offset (for use when zooming)	INT32	---
	Index position of preamble	INT32	---
	Index position of sync word	INT32	---
	Index position of PDU header	INT32	---
	Index position of PDU length field	INT32	---
	Index position of PDU payload	INT32	---
	Index position of CRC start	INT32	---
	Index position of CRC stop	INT32	---

Notes:

1. The index fields contain offsets from the start of the deviation v symbol array that identify the positions of specific fields. A typical use is for plotting deviation v symbol with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.
4. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
5. Binary data will always be a multiple of 4 bytes.

6. Refer to sections 5.3.3 and 5.3.4 *Binary_Header_Information* and *Binary_Data_Fields*.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BLE:DEV? 1,11
```

:FETCh:SRWireless:PACKet:BLE:IBEMissions?

Function

Fetch Bluetooth LE in-band emissions measurements for 1 Mbps and 2 Mbps (2LE) (TRM-LE/CA/BV-03-C and TRM-LE/CA/BV-08-C). Measurements on BLR signals are also supported although test limits are not defined.

To return a full set of results for 2LE measurements an extension code is required.

Extension code bit	Additional fields in response
0	Additional results at ±6, 7 and 8 MHz from the transmit frequency as required when measuring 2LE signals.

Notes:
The following table defines which of the response fields contain valid results, depending on whether the signal being measured is BLE (1 Mbps), 2LE (2 Mbps) or BLR. The value NaN, 'not a number' is defined as 9.91E+37.

<table>
<tr><th></th><th colspan="3">Bluetooth Low Energy Packet Type</th></tr>
<tr><th>Offset from TX frequency</th><th>(Original) BLE
1 Msymbols/s</th><th>2LE
2 Msymbols/s</th><th>BLR
Long Range</th></tr>
<tr><td>±8 MHz</td><td rowspan="3">NaN returned for power measurements and “PASS” returned for limit tests.</td><td rowspan="3">Power results are returned and checked against limits.</td><td rowspan="3">NaN returned for power measurements and “PASS” results for limit tests.</td></tr>
<tr><td>±7 MHz</td></tr>
<tr><td>±6 MHz</td></tr>
<tr><td>±5 MHz</td><td rowspan="4">Power results are returned and checked against limits.</td><td rowspan="2">Power results are returned and checked against limits</td><td rowspan="6">Power results are returned but limit results are always “PASS” because the Bluetooth standard does not specify limits for BLR signals.</td></tr>
<tr><td>±4 MHz</td></tr>
<tr><td>±3 MHz</td><td rowspan="4">Power results are returned but limit results are always “PASS” because the Bluetooth standard does not specify limits at these frequency offsets.</td></tr>
<tr><td>±2 MHz</td></tr>
<tr><td>±1 MHz</td><td rowspan="2">Power results are returned but limit results are always “PASS” because the Bluetooth standard does not specify limits at these frequency offsets.</td></tr>
<tr><td>0 Hz</td></tr>
</table>

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BLE:IBEMissions?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code	Integer	---	---
	If bit 0 of extension code is set:			
4	TX frequency – 8 MHz power	Real number	dBm	0.01 dB
5	TX frequency – 7 MHz power	Real number	dBm	0.01 dB
6	TX frequency – 6 MHz power	Real number	dBm	0.01 dB
7	TX frequency – 5 MHz power	Real number	dBm	0.01 dB
8	TX frequency – 4 MHz power	Real number	dBm	0.01 dB
9	TX frequency – 3 MHz power	Real number	dBm	0.01 dB
10	TX frequency – 2 MHz power	Real number	dBm	0.01 dB
11	TX frequency – 1 MHz power	Real number	dBm	0.01 dB
12	TX frequency power	Real number	dBm	0.01 dB
13	TX frequency + 1 MHz power	Real number	dBm	0.01 dB
14	TX frequency + 2 MHz power	Real number	dBm	0.01 dB
15	TX frequency + 3 MHz power	Real number	dBm	0.01 dB
16	TX frequency + 4 MHz power	Real number	dBm	0.01 dB
17	TX frequency + 5 MHz power	Real number	dBm	0.01 dB
	If bit 0 of extension code is set:			
18	TX frequency + 6 MHz power	Real number	dBm	0.01 dB
19	TX frequency + 7 MHz power	Real number	dBm	0.01 dB
20	TX frequency + 8 MHz power	Real number	dBm	0.01 dB
21	Overall test passed	Boolean	---	---
	If bit 0 of extension code is set:			
22	TX frequency – 8 MHz test passed	Boolean	---	---
23	TX frequency – 7 MHz test passed	Boolean	---	---
24	TX frequency – 6 MHz test passed	Boolean	---	---
25	TX frequency – 5 MHz test passed	Boolean	---	---
26	TX frequency – 4 MHz test passed	Boolean	---	---
27	TX frequency – 3 MHz test passed	Boolean	---	---
28	TX frequency – 2 MHz test passed	Boolean	---	---
29	TX frequency – 1 MHz test passed	Boolean	---	---
30	TX frequency test passed	Boolean	---	---
31	TX frequency + 1 MHz test passed	Boolean	---	---
32	TX frequency + 2 MHz test passed	Boolean	---	---
33	TX frequency + 3 MHz test passed	Boolean	---	---
34	TX frequency + 4 MHz test passed	Boolean	---	---
35	TX frequency + 5 MHz test passed	Boolean	---	---

5 Remote Operation

	If bit 0 of extension code is set:			
36	TX frequency + 6 MHz test passed	Boolean	---	---
37	TX frequency + 7 MHz test passed	Boolean	---	---
38	TX frequency + 8 MHz test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number

Example of Use

```
:FETC:SRW:PACK:BLE:IBEM? 1,3,1
```

:FETCh:SRWireless:PACKet:BLE:MODulation?

Function

Fetch *Bluetooth* LE measurements - modulation characteristics (TRM-LE/CA/BV-05-C, TRM-LE/CA/BV-09-C, TRM-LE/CA/BV-10-C, TRM-LE/CA/BV-11-C, and TRM-LE/CA/BV-13-C)

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BLE:MODulation?
```

Response

The following responses apply for measurements on BLE (1 Mbps), 2LE (2 Mbps) and BLR S=2 packets:

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Δf2 avg	Real number	Hz	100 Hz
4	Δf2 max (	Real number	Hz	100 Hz
5	Δf1 avg	Real number	Hz	100 Hz
6	Δf1 max	Real number	Hz	100 Hz
7	Δf2 avg / Δf1 avg	Real number	---	0.01
8	%Δf2 max > 185 kHz	Real number	%	0.1 %
9	Δf1 avg lower limit test passed	Boolean	---	---
10	Δf1 avg upper limit test passed	Boolean	---	---
11	Δf2 avg / Δf1 avg limit test passed	Boolean	---	---
12	%Δf2 max > 185 kHz limit test passed	Boolean	---	---

The following responses apply for measurements on BLR S=8 packets:

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	NaN (Δf2 measurements are not possible on BLR S=8 packets)	Real number		
4	Δf1 max – lowest value measured in packet.	Real number	Hz	100 Hz
5	Δf1 avg	Real number	Hz	100 Hz
6	Δf1 max – highest value measured in packet	Real number	Hz	100 Hz
7	NaN (Δf2 avg / Δf1 avg is not applicable for BLR S=8)	Real number	---	0.01
8	%Δf1 max > 185 kHz	Real number	%	0.1 %
9	Δf1 avg lower limit test passed	Boolean	---	---
10	Δf1 avg upper limit test passed	Boolean	---	---

11	0 (false) (Δf2 avg / Δf1 avg test is not performed for BLR S=8)	Boolean	---	---
12	%Δf1 max > 115 kHz limit test passed	Boolean	---	---

Notes on SIG Compliant and Speed Test modes (see :CONFigure:SRWireless:BLE:MODE):

1. If the mode is SPEED, all response fields will be populated with results (but see the notes for BLR S=8 packets below).
2. If the mode is STANDARD, some response fields will be populated with results depending on whether the measurement can be made with the specified packet.
3. If a measurement cannot be made in STANDARD mode, the measurement field will contain the value NAN, 'not a number' (defined as 9.91E+37), any associated Boolean flags will be set to 0 (false), and any associated packet number will be set to zero.

Notes on measurements on BLR S=2 and S=8 packets:

1. TRM-LE/CA/BV-12-C specifies measurements on BLR S=8 packets having a 11111111 payload which, after encoding, produces a repeating 00111100 pattern suitable for measurement. In STANDARD mode BLR S=2 packets and BLR S=8 packets containing a payload other than 11111111 will not return results.
2. BLR S=2 and S=8 packets will return measurements in SPEED mode.
3. BLR S=8 packet do not include 0101 patterns, so "Δf2" measurements are not returned.
4. If a measurement cannot be made the measurement field will contain the value NaN, 'not a number' (defined as 9.91E+37), any associated Boolean flags will be set to 0 (false), and any associated packet number (for example 'Packet number for Δf2 max worst case') will be set to zero.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BLE:MOD? 1,4
```

:FETCh:SRWireless:PACKet:BLE:POWer?

Function

Fetch *Bluetooth* LE measurements - output power (TRM-LE/CA/BV-01-C).

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BLE:POWer?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Average power	Real number	dBm	0.01 dB
4	Crest factor	Real number	dB	0.01 dB
5	Average power lower limit test passed	Boolean	---	---
6	Average power upper limit test passed	Boolean	---	---
7	Crest factor test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BLE:POW? 1,6
```

:FETCh:SRWireless:PACKet:BLE:PPRofile:AVERage:ASCii?

Function

Fetch *Bluetooth* measurements - power burst profile (BLE) – averaged data – ASCII format

Command

N/A

Query

`:FETCh:SRWireless:PACKet:BLE:PPRofile:AVERage:ASCii?`

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	
2	Packet index	Integer	---	
3	Start time (relative to p0)	Real number	s	1 µs
4	Stop time (relative to p0)	Real number	s	1 µs
5	Number of points (n)	Integer	---	
6	First average power profile point	Real number	dBm	0.01 dB
---	…			
	Last average power profile point	Real number	dBm	0.01 dB
	Index fields valid	Boolean	---	---
	Start offset	Integer	---	---
	Stop offset	Integer	---	---
	Preamble index (p0)	Integer	---	---
	Sync word index	Integer	---	---
	PDU header index	Integer	---	---
	PDU length index	Integer	---	---
	PDU payload index	Integer	---	---
	CRC start index	Integer	---	---
	CRC stop index	Integer	---	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile. A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BLE:PPR:AVER:ASC? 1,11
```

:FETCh:SRWireless:PACKet:BLE:PPRofile:AVERage[:BINary]?

Function

Fetch *Bluetooth* measurements - power burst profile (BLE) – average data – binary format.

Command

N/A

Query

`:FETCh:SRWireless:PACKet:BLE:PPRofile:AVERage[:BINary]?`

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Start time (relative to p0)	REAL32	s
5	Stop time (relative to p0)	REAL32	s
6	Number of points (n)	INT32	---
	Power in dBm	REAL32	dBm
	Index fields valid if non zero.	INT32	---
	Start offset (for use when zooming)	INT32	---
	Stop offset (for use when zooming)	INT32	---
	Index position of preamble	INT32	---
	Index position of sync word	INT32	---
	Index position of PDU header	INT32	---
	Index position of PDU length field	INT32	---
	Index position of PDU payload	INT32	---
	Index position of CRC start	INT32	---
	Index position of CRC stop	INT32	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile. A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.
4. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
5. Binary data will always be a multiple of 4 bytes.
6. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BLE:PPR:AVER? 1,10
```

:FETCh:SRWireless:PACKet:BLE:PPRofile:PEAK:ASCii?

Function

Fetch *Bluetooth* measurements - power burst profile (BLE) – peak data – ASCII format

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BLE:PPRofile:PEAK:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	
2	Packet index	Integer	---	
3	Start time (relative to p0)	Real number	s	1 μs
4	Stop time (relative to p0)	Real number	s	1 μs
5	Number of points (n)	Integer	---	
6	First peak power profile point	Real number	dBm	0.01 dB
---	...			
	Last peak power profile point	Real number	dBm	0.01 dB
	Index fields valid	Boolean	---	---
	Start offset	Integer	---	---
	Stop offset	Integer	---	---
	Preamble index (p0)	Integer	---	---
	Sync word index	Integer	---	---
	PDU header index	Integer	---	---
	PDU length index	Integer	---	---
	PDU payload index	Integer	---	---
	CRC start index	Integer	---	---
	CRC stop index	Integer	---	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile. A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BLE:PPR:PEAK:ASC? 1,4
```

:FETCh:SRWireless:PACKet:BLE:PPRofile:PEAK[:BINary]?

Function

Fetch *Bluetooth* measurements - power burst profile (BLE) – peak data – binary format.

Command

N/A

Query

`:FETCh:SRWireless:PACKet:BLE:PPRofile:PEAK[:BINary]?`

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Start time (relative to p0)	REAL32	s
5	Stop time (relative to p0)	REAL32	s
6	Number of points (n)	INT32	---
7	Power in dBm	REAL32	dBm
…	Index fields valid if non zero.	INT32	
	Start offset (for use when zooming)	INT32	---
	Stop offset (for use when zooming)	INT32	---
	Index position of preamble	INT32	---
	Index position of sync word	INT32	---
	Index position of PDU header	INT32	---
	Index position of PDU length field	INT32	---
	Index position of PDU payload	INT32	---
	Index position of CRC start	INT32	---
	Index position of CRC stop	INT32	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile. A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.
4. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
5. Binary data will always be a multiple of 4 bytes.
6. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BLE:PPR:PEAK? 1,7
```

:FETCh:SRWireless:PACKet:BT:ACPower?

Function

Fetch the *Bluetooth* adjacent channel power measurements (TRM/CA/06/C).

Command

N/A

Query

`:FETCh:SRWireless:PACKet:BT:ACPower?`

Response

#	Description	Format	Units
1	Segment index	Integer	---
2	Packet index	Integer	---
3	Channel–5 power	Real Number	dBm
4	Channel–4 power	Real Number	dBm
5	Channel–3 power	Real Number	dBm
6	Channel–2 power	Real Number	dBm
7	Channel–1 power	Real Number	dBm
8	Channel power	Real Number	dBm
9	Channel+1 power	Real Number	dBm
10	Channel+2 power	Real Number	dBm
11	Channel+3 power	Real Number	dBm
12	Channel+4 power	Real Number	dBm
13	Channel+5 power	Real Number	dBm
14	Overall test result	Boolean	---
15	Channel–5 test passed	Boolean	---
16	Channel–4 test passed	Boolean	---
17	Channel–3 test passed	Boolean	---
18	Channel–2 test passed	Boolean	---
19	Channel–1 test passed (always pass)	Boolean	---
20	Channel test passed (always pass)	Boolean	---
21	Channel+1 test passed (always pass)	Boolean	---
22	Channel+2 test passed	Boolean	---
23	Channel+3 test passed	Boolean	---
24	Channel+4 test passed	Boolean	---
25	Channel+5 test passed	Boolean	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:ACP? 1,3
```

:FETCh:SRWireless:PACKet:BT:BANDwidth?

Function

Fetch *Bluetooth* 20 dB bandwidth measurement results (TRM/CA/05/C).
Currently no extension codes have been defined for this command. Use zero.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:BANDwidth?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	20 dB bandwidth	Integer	Hz	1 Hz
5	Bandwidth limit test passed	Boolean	---	---
6	Spectrum bandwidth (in practice always 3 MHz).	Integer	Hz	1 Hz
7	Number of points in spectrum	Integer	---	---
	For each point in spectrum:	Integer	---	---
1	Power (averaged over segment)	Real number	dBm	0.01 dB

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number

Example of Use

```
:FETC:SRW:PACK:BT:BAND? 1,1,0
>1,1,0,914062,1,3000000,384,-70.39,-70.97,…      // Segment 1
                                                  // Packet 1
                                                  // Extension code 0
                                                  // 20 dB BW 914.062 kHz
                                                  // Limit test passed.
                                                  // Spectrum BW is 3 MHz
                                                  // 384 points in spectrum
                                                  // Power values…
```

:FETCh:SRWireless:PACKet:BT:CDRift?

Function

Fetch *Bluetooth* measurements - carrier drift (TRM/CA/09/C)

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:CDRift?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Carrier frequency drift	Real number	Hz	Hz
4	Carrier frequency drift rate	Real number	Hz	Hz
5	Carrier frequency drift limit test passed	Boolean	---	---
6	Carrier frequency drift rate test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:CDR? 1,10
```

:FETCh:SRWireless:PACKet:BT:CONSt[:BINary]?

Function

Fetch *Bluetooth* measurements - constellation (EDR) – binary format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:CONSt[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm… mm	Binary header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Number of points (IQ pairs)	INT32	---
5	First I value	REAL32	---
6	First Q value	REAL32	---
…	…	…	…
	Last I value	REAL32	---
	Last Q value	REAL32	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Note:

1. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Example of Use

```
:FETC:SRW:PACK:BT:CONS? 1,4
```

:FETCh:SRWireless:PACKet:BT:DEViation:ASCii?

Function

Fetch *Bluetooth* measurements - deviation vs symbol (BR) – ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:DEViation:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	
2	Packet index	Integer	---	
3	Start time	Real number	s	1 µs
4	Stop time	Real number	s	1 µs
5	Number of points (n)	Integer	---	
6	First deviation point	Real number	Hz	1 Hz
---	…			
	Last deviation point	Real number	Hz	1 Hz
	Index fields valid	Boolean	---	---
	Start offset	Integer	---	---
	Stop offset	Integer	---	---
	Preamble index (p0)	Integer	---	---
	Sync word index	Integer	---	---
	Trailer index	Integer	---	---
	Packet header index	Integer	---	---
	Guard time index	Integer	---	---
	EDR sync index	Integer	---	---
	Payload header index	Integer	---	---
	Payload index	Integer	---	---
	CRC start index	Integer	---	---
	CRC stop index	Integer	---	---

Notes:

1. The index fields contain offsets from the start of the deviation v symbol array that identify the positions of specific fields. A typical use is for plotting deviation v symbol with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:DEV:ASC? 1,3
```

:FETCh:SRWireless:PACKet:BT:DEViation[:BINary]?

Function

Fetch *Bluetooth* measurements - deviation vs symbol (BR) – binary format

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:DEViation[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm...mm	Binary header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Start time (relative to p0)	REAL32	s
5	Stop time (relative to p0)	REAL32	s
6	Number of points (n)	INT32	---
7	Position of p0 relative to start of power data	INT32	---
8	Deviation in Hz	REAL32	Hz
...	...		
	Last deviation point	REAL32	Hz
	Index fields (valid if non-zero)	INT32	---
	Start offset	INT32	---
	Stop offset	INT32	---
	Preamble index (p0)	INT32	---
	Sync word index	INT32	---
	Trailer index	INT32	---
	Packet header index	INT32	---
	Guard time index	INT32	---
	EDR sync index	INT32	---
	Payload header index	INT32	---
	Payload index	INT32	---
	CRC start index	INT32	---
	CRC stop index	INT32	---

Notes:

1. The index fields contain offsets from the start of the deviation v symbol array that identify the positions of specific fields. A typical use is for plotting deviation v symbol with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain

zero.

4. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
5. Binary data will always be a multiple of 4 bytes.
6. Refer to sections 5.3.3 and 5.3.4 *Binary_Header_Information* and *Binary_Data_Fields*.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:DEV? 1,11
```

:FETCh:SRWireless:PACKet:BT:DEVM:ASCii?

Function

Fetch *Bluetooth* measurements - DEVM v symbol (EDR) – ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:DEVM:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Number of points	Integer	---	---
4	First DEVM value	Real number	---	0.001
…				
	Last DEVM value	Real number	---	0.001

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:DEVM:ASC? 1,10
```

:FETCh:SRWireless:PACKet:BT:DEVM[:BINary]?

Function

Fetch *Bluetooth* measurements - DEVM v symbol (EDR) – binary format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:DEVM[:BINary]?
```

Response

#	Description	Format	Units	Resolution
1	#nmmm…mm	Binary header	---	---
2	Segment index	INT32	---	---
3	Packet index	INT32	---	---
4	Number of points	INT32	---	---
5	First DEVM value	REAL32	---	0.01
…	…	…	…	…
	Last DEVM value	REAL32	---	0.01

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Note:

1. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Example of Use

```
:FETC:SRW:PACK:BT:DEVM? 1,9
```

:FETCh:SRWireless:PACKet:BT:ECModulation?

Function

Fetch *Bluetooth* measurements - EDR carrier frequency stability and modulation accuracy (TRM/CA/11/C)

Command

N/A

Query

`:FETCh:SRWireless:PACKet:BT:ECModulation`

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer		---
3	Initial frequency error (ω_i)	Real number	Hz	1 Hz
4	Block Frequency error (ω_0)	Real number	Hz	1 Hz
5	Block + initial frequency error ($\omega_i + \omega_0$)	Real number	Hz	1 Hz
6	Payload modulation scheme	See definition	---	---
7	RMS DEVM	Real number		0.001
8	Peak DEVM	Real number		0.001
9	% symbols meeting DEVM limit	Real number		0.01
10	Guard interval	Real number	s	50 ns
11	Carrier frequency stability limit test passed	Boolean	---	---
12	RMS DEVM limit test passed	Boolean	---	---
13	Peak DEVM limit test passed	Boolean	---	---
14	Percentage of symbols limit test passed	Boolean	---	---
15	Guard interval limit test passed	Boolean	---	---

Payload modulation scheme:

PI4	$\pi/4$ DQPSK
8DPSK	8 DPSK

Notes on SIG Compliant and Speed Test modes (see :CONFigure:SRWireless:BT:MODE):

1. If the mode is SPEED, all response fields will be populated with results.
2. If the mode is STANDARD, some response fields will be populated with results depending on whether the measurement can be made with the specified packet payload according to the SIG Specification.
3. If a measurement cannot be made in STANDARD mode, the measurement field will contain the value NAN, 'not a number' (defined as 9.91E+37), any associated Boolean flags will be set to 0 (false), and any associated packet number (for example 'Packet

number for Δf2 max worst case') will be set to zero.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:ECM? 1,7
```

:FETCh:SRWireless:PACKet:BT:ECModulation? (Extended)

Function

Fetch *Bluetooth* measurements - EDR carrier frequency stability and modulation accuracy (TRM/CA/11/C). The optional extension code adds additional results to the response as described in the following table:

Extension code bit	Additional fields in response
0	Individual limit test results for: • Initial frequency error (ω_i) • Frequency error (ω_0) • Block frequency error ($\omega_i + \omega_0$)

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:ECModulation?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code	Integer	---	
4	Initial frequency error (ω_i)	Real number	Hz	1 Hz
5	Frequency error (ω_0)	Real number	Hz	1 Hz
6	Block frequency error ($\omega_i + \omega_0$)	Real number	Hz	1 Hz
7	Payload modulation scheme	<CRD>	---	---
8	RMS DEVM	Real number		0.001
9	Peak DEVM	Real number		0.001
10	% symbols meeting DEVM limit	Real number	%	0.01 %
11	Guard interval	Real number	s	50 ns

	If bit 0 of extension code set:			
12	Initial frequency error (ω_i) limit test passed	Boolean	---	---
13	Frequency error (ω_0) limit test passed	Boolean	---	---
14	Block frequency error ($\omega_i + \omega_0$) limit test passed	Boolean	---	---
15	Carrier frequency stability limit test passed **Note:** *this is the logical AND of the limit test results for (ω_i), (ω_0) and ($\omega_i + \omega_0$).*	Boolean	---	---
16	RMS DEVM limit test passed	Boolean	---	---
17	Peak DEVM limit test passed	Boolean	---	---
18	Percentage of symbols limit test passed	Boolean	---	---
19	Guard interval limit test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number

Example of Use

```
:FETC:SRW:PACK:BT:ECM? 1,7,1
```

:FETCh:SRWireless:PACKet:BT:EDPencoding?

Function

Fetch the *Bluetooth* EDR differential phase encoding measurements (TRM/CA/12/C).

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:EDPencoding?
```

Response

#	Description	Format	Units
1	Segment index	Integer	---
2	Packet index (always zero)	Integer	---
3	Extension code (currently zero)	Integer	---
4	Bit error rate (BER)	Real Number	%
5	Number of bit errors	Integer	---
6	Number of bits tested	Integer	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code (currently zero)	Number

Example of Use

```
:FETC:SRW:PACK:BT:EDP? 1,7,0
```

:FETCh:SRWireless:PACKet:BT:ERTPower?

Function

Fetch *Bluetooth* measurements - EDR relative TX power (TRM/CA/10/C)

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:ERTPower?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	P_FSK	Real number	dBm	0.01 dB
4	P_PSK	Real number	dBm	0.01 dB
5	P_PSK - P_FSK	Real number	dB	0.01 dB
6	Relative transmit power limit test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:ERTP? 1,4
```

:FETCh:SRWireless:PACKet:BT:EYE:ASCii?

Function

Fetch *Bluetooth* measurements - eye diagram (BR / BLE) – ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:EYE:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Symbol length (oversampling factor)	Integer	---	---
4	GFSK data start: offset relative to p0	Integer	---	---
5	GFSK data stop: offset relative to p0	Integer	---	---
6	Number of points	Integer	---	---
7	First frequency deviation value	Real number	Hz	1 Hz
…	…			
	Last frequency deviation value	Real number	Hz	1 Hz

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:EYE:ASC? 1,15
```

:FETCh:SRWireless:PACKet:BT:EYE[:BINary]?

Function

Fetch *Bluetooth* measurements - eye diagram (BR / BLE) – binary format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:EYE[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm...mm	Binary header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Symbol length (oversampling factor)	INT32	---
5	GFSK data start: offset relative to p0	INT32	---
6	GFSK data stop: offset relative to p0	INT32	---
7	Number of points	INT32	---
8	First frequency deviation value	REAL32	Hz
...	...		
	Last frequency deviation value	REAL32	Hz

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Note:

Refer to sections 5.3.3 and 5.3.4 *Binary_Header_Information* and *Binary_Data_Fields*.

Example of Use

```
:FETC:SRW:PACK:BT:EYE? 1,7
```

:FETCh:SRWireless:PACKet:BT:FRANge?

Function

Fetch Bluetooth frequency range measurements (TRM/CA/04/C).
Currently no extension codes have been defined for this command. Use zero.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:FRANge?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Center frequency	Integer	Hz	1 Hz
5	Peak power	Real number	dBm	0.01 dB
6	Low frequency (f_L)	Integer	Hz	1 Hz
7	High frequency (f_H)	Integer	Hz	1 Hz
8	Low frequency limit test passed	Boolean	---	---
9	High frequency limit test passed	Boolean	---	---
10	Spectrum bandwidth	Integer	Hz	1 Hz
11	Number of points	Integer	---	---
12	First power value	Real number	dBm	0.01 dB
…	…			
	Last power value	Real number	dBm	0.01 dB

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number

Example of Use

```
:FETC:SRW:PACK:BT:FRAN? 1,7,0
```

:FETCh:SRWireless:PACKet:BT:IBSPurious?

Function

Fetch the *Bluetooth* EDR in-band spurious emissions measurements (TRM/CA/13/C).

Command

N/A

Query

`:FETCh:SRWireless:PACKet:BT:IBSPurious?`

Response

#	Description	Format	Units
1	Segment index	Integer	---
2	Packet index	Integer	---
3	Channel–5 power	Real Number	dBm
4	Channel–4 power	Real Number	dBm
5	Channel–3 power	Real Number	dBm
6	Channel–2 power	Real Number	dBm
7	Channel–1 power	Real Number	dBm
8	Channel power	Real Number	dBm
9	Channel+1 power	Real Number	dBm
10	Channel+2 power	Real Number	dBm
11	Channel+3 power	Real Number	dBm
12	Channel+4 power	Real Number	dBm
13	Channel+5 power	Real Number	dBm
14	Overall test result	Boolean	---
15	Channel–5 test passed	Boolean	---
16	Channel–4 test passed	Boolean	---
17	Channel–3 test passed	Boolean	---
18	Channel–2 test passed	Boolean	---
19	Channel–1 test passed (always pass)	Boolean	---
20	Channel test passed (always pass)	Boolean	---
21	Channel+1 test passed (always pass)	Boolean	---
22	Channel+2 test passed	Boolean	---
23	Channel+3 test passed	Boolean	---
24	Channel+4 test passed	Boolean	---
25	Channel+5 test passed	Boolean	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

5 Remote Operation

Example of Use

```
:FETC:SRW:PACK:BT:IBSP? 1,2
```

:FETCh:SRWireless:PACKet:BT:ICFTolerance?

Function

Fetch *Bluetooth* measurements - initial carrier frequency tolerance (TRM/CA/08/C).

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:ICFTolerance?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Initial carrier frequency tolerance	Real number	Hz	1 Hz
4	Initial carrier frequency tolerance limit test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:ICFT? 1,1
```

:FETCh:SRWireless:PACKet:BT:MODulation?

Function

Fetch *Bluetooth* measurements - modulation characteristics (TRM/CA/07/C).

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:MODulation?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Δf2 avg	Real Number	Hz	100 Hz
4	Δf2 max	Real Number	Hz	100 Hz
5	Δf1 avg	Real Number	Hz	100 Hz
6	Δf1 max	Real Number	Hz	100 Hz
7	Δf2 avg / Δf1 avg	Real Number	---	0.01
8	%Δf2 max > 115 kHz	Real Number	%	0.01 %
9	Δf1 avg lower limit test passed	Boolean	---	---
10	Δf1 avg upper limit test passed	Boolean	---	---
11	Δf2 avg / Δf1 avg limit test passed	Boolean	---	---
12	%Δf2 max > 115 kHz limit test passed	Boolean	---	---

Notes on SIG Compliant and Speed Test modes (see :CONFigure:SRWireless:BT:MODE):

1. If the mode is SPEED, all response fields will be populated with results.
2. If the mode is STANDARD, some response fields will be populated with results depending on whether the measurement can be made with the specified packet payload according to the SIG Specification.
3. If a measurement cannot be made in STANDARD mode, the measurement field will contain the value NAN, 'not a number' (defined as 9.91E+37), any associated Boolean flags will be set to 0 (false), and any associated packet number (for example 'Packet number for Δf2 max worst case') will be set to zero.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:MOD? 1,5
```

:FETCh:SRWireless:PACKet:BT:POWer?

Function

Fetch *Bluetooth* measurements - transmit power (TRM/CA/01/C).

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:POWer?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Average power	Real number	dBm	0.01 dB
4	Peak power	Real number	dBm	0.01 dB
5	Average power lower limit test passed	Boolean	---	
6	Average power upper limit test passed	Boolean	---	
7	Peak power test passed	Boolean	---	

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:POW? 1,1
```

:FETCh:SRWireless:PACKet:BT:PPRofile:AVERage:ASCii?

Function

Fetch *Bluetooth* measurements - power burst profile (BR / EDR) – averaged data – ASCII format

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:PPRofile:AVERage:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	
2	Packet index	Integer	---	
3	Start time (relative to p0)	Real number	s	1 μs
4	Stop time (relative to p0)	Real number	s	1 μs
5	Number of points (n)	Integer	---	
6	First average power profile point	Real number	dBm	0.01 dB
---	…			
	Last average power profile point	Real number	dBm	0.01 dB
	Index fields valid	Boolean	---	---
	Start offset	Integer	---	---
	Stop offset	Integer	---	---
	Preamble index (p0)	Integer	---	---
	Sync word index	Integer	---	---
	Trailer index	Integer	---	---
	Packet header index	Integer	---	---
	Guard time index	Integer	---	---
	EDR sync index	Integer	---	---
	Payload header index	Integer	---	---
	Payload index	Integer	---	---
	CRC start index	Integer	---	---
	CRC stop index	Integer	---	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile. A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:PPR:AVER:ASC? 1,11
```

5

Remote Operation

:FETCh:SRWireless:PACKet:BT:PPRofile:AVERage[:BINary]?

Function

Fetch *Bluetooth* measurements - power burst profile (BR / EDR) – average data – binary format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:PPRofile:AVERage[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Start time (relative to p0)	REAL32	s
5	Stop time (relative to p0)	REAL32	s
6	Number of points (n)	INT32	---
7	First average power profile point	REAL32	dBm
…	…		
	Last average power profile point	REAL32	dBm
	Index fields (valid if non-zero)	INT32	---
	Start offset	INT32	---
	Stop offset	INT32	---
	Preamble index (p0)	INT32	---
	Sync word index	INT32	---
	Trailer index	INT32	---
	Packet header index	INT32	---
	Guard time index	INT32	---
	EDR sync index	INT32	---
	Payload header index	INT32	---
	Payload index	INT32	---
	CRC start index	INT32	---
	CRC stop index	INT32	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile. A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.

3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.
4. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
5. Binary data will always be a multiple of 4 bytes.
6. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:PPR:AVER? 1,10
```

:FETCh:SRWireless:PACKet:BT:PPRofile:PEAK:ASCii?

Function

Fetch *Bluetooth* measurements - power burst profile (BR / EDR) – peak data – ASCII format

Command

N/A

Query

`:FETCh:SRWireless:PACKet:BT:PPRofile:PEAK:ASCii?`

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	
2	Packet index	Integer	---	
3	Start time (relative to p0)	Real number	s	1 µs
4	Stop time (relative to p0)	Real number	s	1 µs
5	Number of points (n)	Integer	---	
6	First peak power profile point	Real number	dBm	0.01 dB
---	…			
	Last peak power profile point	Real number	dBm	0.01 dB
	Index fields valid	Boolean	---	---
	Start offset	Integer	---	---
	Stop offset	Integer	---	---
	Preamble index (p0)	Integer	---	---
	Sync word index	Integer	---	---
	Trailer index	Integer	---	---
	Packet header index	Integer	---	---
	Guard time index	Integer	---	---
	EDR sync index	Integer	---	---
	Payload header index	Integer	---	---
	Payload index	Integer	---	---
	CRC start index	Integer	---	---
	CRC stop index	Integer	---	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile. A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:PPR:PEAK:ASC? 1,4
```

:FETCh:SRWireless:PACKet:BT:PPRofile:PEAK[:BINary]?

Function

Fetch *Bluetooth* measurements - power burst profile (BR / EDR) – peak data – binary format.

Command

N/A

Query

`:FETCh:SRWireless:PACKet:BT:PPRofile:PEAK[:BINary]?`

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Start time (relative to p0)	REAL32	s
5	Stop time (relative to p0)	REAL32	s
6	Number of points (n)	INT32	---
7	First peak power profile point	REAL32	dBm
…	…		
	Last peak power profile point	REAL32	dBm
	Index fields (valid if non-zero)	INT32	---
	Start offset	INT32	---
	Stop offset	INT32	---
	Preamble index (p0)	INT32	---
	Sync word index	INT32	---
	Trailer index	INT32	---
	Packet header index	INT32	---
	Guard time index	INT32	---
	EDR sync index	INT32	---
	Payload header index	INT32	---
	Payload index	INT32	---
	CRC start index	INT32	---
	CRC stop index	INT32	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile. A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.

3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.
4. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
5. Binary data will always be a multiple of 4 bytes.
6. Refer to sections 5.3.3 and 5.3.4 *Binary_Header_Information* and *Binary_Data_Fields*.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:PPR:PEAK? 1,7
```

:FETCh:SRWireless:PACKet:BT:SPECtrum:ASCii?

Function

Fetch *Bluetooth* measurements - spectrum (BR / EDR) – ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:SPECtrum:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Bandwidth		Hz	1 Hz
4	Number of points in spectrum	Integer	---	---
5	First power value	Real number	dBm	0.01 dB
	…			
	Last power value	Real number	dBm	0.01 dB

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:SPEC:ASC? 1,3
```

:FETCh:SRWireless:PACKet:BT:SPECtrum[:BINary]?

Function

Fetch *Bluetooth* measurements - spectrum (BR / EDR) – binary format

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:SPECtrum[:BINary]?
```

Response

#	Description	Format	Units
1	#nmmm… mm	Binary header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Bandwidth	REAL32	Hz
5	Number of points	INT32	---
6	First power value	REAL32	dBm
…	…		
	Last power value	REAL32	dBm

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Note:

1. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Example of Use

```
:FETC:SRW:PACK:BT:SPEC? 1,2
```

5

Remote Operation

:FETCh:SRWireless:PACKet:BT:VECTor[:BINary]?

Function

Fetch *Bluetooth* measurements - vector diagram (EDR).

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:BT:VECTor[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm… mm	Binary header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Basic rate start index	INT32	---
5	Basic rate stop index	INT32	---
6	EDR start index	INT32	---
7	EDR stop index	INT32	---
8	Number of points (IQ pairs)	INT32	---
9	First I value	REAL32	---
10	First Q value	REAL32	---
…	…	…	…
	Last I value	REAL32	---
	Last Q value	REAL32	---

Notes:

1. The basic rate and EDR start and stop indices allow the BR and EDR parts of the IQ array to be identified. A typical use is to plot these as separate diagrams or in different colors.
2. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:BT:VECT? 1,5
```

:FETCh:SRWireless:PACKet:IDENtity?

Function

Fetch information about individual packets that have been captured and identified automatically. Following capture of a segment where the wireless standard is AUTODSSS, AUTOOFDM or AUTOBT, this command may be used to read information about each individual packet.

Note:
This query returns a similar response to :FETCh:SRWireless:SEGMent:IDENtity? The difference between the two commands is that :FETCh:SRWireless:SEGMent:IDENtity? returns information about the **first** packet identified. That packet type then becomes a reference. All other packets in the segment are compared against the reference, and only if they match are their measurement results averaged and made available using the ":FETCh:SRWireless:SUMMary…?" commands. :FETCh:SRWireless:PACKet:IDENtity? may be used to extract information from **any** packet within a segment for which results are available.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:IDENtity?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code	Integer	---	---
4	Segment status	Integer	---	---
5	Decoded standard	See definition	---	---
	If decoded standard is Bluetooth / BLE			
6	Bluetooth address (LAP)	Hexadecimal	---	---
7	Bluetooth packet type	See definition	---	---
8	Bluetooth payload type	See definition	---	---
9	Payload length	Integer	---	---
	If bit 0 of the extension code is set:			
10	Packet header LT address	Integer	---	---
11	Packet header type as numeric value	Integer	---	---
12	Packet header flow bit	Integer	---	---
13	Packet ARQN bit	Integer	---	---
14	Packet header SEQN bit	Integer	---	---

15	Packet header HEC	Hexadecimal	---	---
16	Payload header LLID	Integer	---	---
17	Payload header flow	Integer	---	---
18	CRC status	See definition	---	---
	If bit 1 of the extension code is set:			
19	Packet type (from analysis of captured data)	See definition		
20	Payload type (from analysis of captured data)	See definition		
21	Payload length (from analysis of captured data)	Integer		
	Note: The above three fields are intended for use with header exclusion mode. If header exclusion mode is **not** enabled these fields will contain duplicates of information obtained from the packet header.			
	If bit 2 of the extension code is set:			
22	BLE data rate	See definition		
23	BLR coding scheme	See definition		
	If bit 3 of the extension code is set:			
24	Coding indicator (2-bit field)	Binary		

	If decoded standard is 802.11b			
6	Data rate	Real number	Mbps	0.1 Mbps
7	PPDU format	See definition	---	---
8	PSDU length	Integer	---	---
9	PLCP CRC status	See definition	---	---
	If decoded standard is 802.11a or 802.11g:			
6	Type of packet	See definition	---	---
7	Data rate	Real number	Mbps	0.1 Mbps
8	Coding rate	See definition	---	---
9	Coding type	See definition	---	---
10	Modulation type	See definition	---	---
11	PSDU length	Integer	---	---
12	L-SIG parity check	See definition	---	---
13	Number of symbols	Integer	---	---
14	Number of tones	Integer	---	---
15	Guard interval	See definition	---	---
16	No. of signals	Integer	---	---
17	No. of spatial streams	Integer	---	---
18	No. of space time streams	Integer	---	---
	If decoded standard is 802.11p			
6	Type of packet (always LEGACY for p)	See definition		
7	Data rate	Real number	Mbps	0.1 Mbps
8	Coding rate	See definition	---	---

9	Coding type	See definition	---	---
10	Modulation type	See definition	---	---
11	PSDU length	Integer	---	---
12	L-SIG parity check	See definition	---	---
13	Number of symbols	Integer	---	---
14	Number of tones	Integer	---	---
15	Guard interval	See definition	---	---
16	No. of signals	Integer	---	---
17	No. of spatial streams	Integer	---	---
18	No. of space time streams	Integer	---	---
19	Bandwidth (5 MHz, 10 MHz or 20 MHz)	Integer	MHz	1 MHz

If decoded standard is 802.11n:

6	Type of packet	See definition	---	---
7	Data rate	Real number	Mbps	0.1 Mbps
8	Coding rate	See definition	---	---
9	Coding type	See definition	---	---
10	Modulation type	See definition	---	---
11	PSDU length	Integer	---	---
12	L-SIG parity check	See definition	---	---
13	Number of symbols	Integer	---	---
14	Number of tones	Integer	---	---
15	Guard interval	See definition	---	---
16	No. of signals	Integer	---	---
17	No. of spatial streams	Integer	---	---
18	No. of space time streams	Integer	---	---
19	MCS	Integer	---	---
20	PPDU type	See definition	---	---
21	PPDU format	See definition	---	---
22	HT-SIG CRC	See definition	---	---

If decoded standard is 802.11ac

6	Type of packet	See definition	---	---
7	Data rate	Real number	Mbps	0.1 Mbps
8	Coding rate	See definition	---	---
9	Coding type	See definition	---	---
10	Modulation type	See definition	---	---
11	PSDU length	Integer	---	---
12	L-SIG parity check	See definition	---	---
13	Number of symbols	Integer	---	---
14	Number of tones	Integer	---	---
15	Guard interval	See definition	---	---
16	No. of signals	Integer	---	---

17	No. of spatial streams	Integer	---	---
18	No. of space time streams	Integer	---	---
19	MCS	Integer	---	---
20	PPDU type	See definition	---	---
21	VHT-SIG CRC	See definition	---	---

Parameters

#	Description	Format	Units	Min	Max	Default
1	Segment index	Integer	---	---	---	---
2	Packet index	Integer	---	---	---	---
3	Extension code	Integer	---	0	2^{32}-1	0

Decoded standard:

WLA	WLAN 802.11a
WLB	WLAN 802.11b
WLG	WLAN 802.11g
WLP	WLAN 802.11p
WLN	WLAN 802.11n
WLAC	WLAN 802.11ac
BT	Bluetooth basic rate and EDR
BLE	Bluetooth low energy

CRC status / parity checks:

PASS

FAIL

Bluetooth packet type:

DH1, DH3, DH5	Basic Rate
2DH1, 2DH3, 2DH5	EDR 2 Mbps
3DH1, 3DH3, 3DH5	EDR 3 Mbps
NULL, POLL,	Link management packets.
BLE, 2LE, BLR	Bluetooth Low Energy
UNKNOWN	None of the above

Bluetooth payload type:

ZEROS	00000000
AAHEX	10101010
FZEROHEX	11110000
PRBS9	Pseudo random binary sequence (9th order)
55HEX	01010101
ZEROFHEX	00001111
PRBS15	Pseudo random binary sequence (15th order)
ONES	11111111

UNKNOWN	None of the above
Bluetooth low energy data rate:	
125KBPS	125 kbps (BLR S=8)
500KBPS	500 kbps (BLR S=2)
1MBPS	1 Mbps (BLE)
2MBPS	2 Mbps (2LE)
BLR coding scheme:	
S2	BLR S=2 coding
S8	BLR S=8 coding
WLAN packet type::	
LEGACY	802.11g, a, p
HT	802.11n
VHT	802.11ac
Coding rate:	
CR12	1/2
CR34	3/4
CR56	5/6
Coding type:	
BCC	Binary convolutional coding
LDPC	Low density parity check
Modulation type:	
BPSK	Binary phase shift keying
QPSK	Quadrature phase shift keying
QAM16	16-QAM (Quadrature amplitude modulation)
QAM64	64-QAM (Quadrature amplitude modulation)
QAM256	256-QAM (Quadrature amplitude modulation)
Guard interval:	
LONG	Long guard interval (800 ns)
SHORT	Short guard interval (400 ns)
PPDU type (802.11n):	
M20	Data is transmitted over a signal with 20 MHz bandwidth
M40	Data is transmitted over two adjacent signals with 20 MHz bandwidth
M40UPPER	Data is transmitted over a signal with 20 MHz bandwidth in the upper sideband of a 40 MHz channel

M40LOWER	Data is transmitted over a signal with 20 MHz bandwidth in the lower sideband of a 40 MHz channel.
M40DUP	Data is transmitted twice using two adjacent signals with 20 MHz bandwidth

PPDU type (802.11ac):

VHT20	Data is transmitted over a signal with 20 MHz bandwidth
VHT40	Data is transmitted over a signal with 40 MHz bandwidth
VHT80	Data is transmitted over a signal with 80 MHz bandwidth
VHT80_80	Data is transmitted over two signals with 80 MHz bandwidth
VHT160	Data is transmitted over a signal with 160 MHz bandwidth

PPDU format (802.11n):

MIXED	HT Mixed
GREENFIELD	HT Greenfield

PPDU Format (802.11b):
LONG
SHORT

Example of Use

```
:FETC:SRW:PACK:IDEN? 1,1,0
> 1,1,0,0,WLG,LEGACY,54.0,CR34,BCC,QAM64,1077,PASS,40,64,LONG,1,1,1
```

:FETCh:SRWireless:PACKet:WLAN:CCDF:ASCii?

Function

Fetch CCDF profile (graphical) measurements - ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:CCDF:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Number of gates	Integer	---	---
	For first gate			
1	Gate number	Integer	---	---
2	Number of points in CCDF profile	Integer	---	---
3	First CCDF value	Real number	%	0.001%
…	…			
	Last CCDF value	Real number	%	0.001%
	…			
	For last gate			
1	Gate number	Integer	---	---
2	Number of points in CCDF profile	Integer	---	---
3	First CCDF value	Real number	%	0.001%
…	…			
	Last CCDF value	Real number	%	0.001%

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:CCDF:ASC? 1,5
```

:FETCh:SRWireless:PACKet:WLAN:CCDF[:BINary]?

Function

Fetch CCDF profile (graphical) measurements - binary format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:CCDF[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Number of gates	INT32	---
	For first gate:		
1	Gate number	INT32	---
2	Number of points	INT32	---
3	First CCDF value	REAL32	%
…	…		
	Last CCDF value	REAL32	%
…			
	For last gate:		
1	Gate number	INT32	---
2	Number of points	INT32	---
3	First CCDF value	REAL32	%
…	…		
	Last CCDF value	REAL32	%

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Note:

1. Refer to sections 5.3.3 and 5.3.4 *Binary_Header_Information* and *Binary_Data_Fields*.

Example of Use

```
:FETC:SRW:PACK:WLAN:CCDF? 1,9
```

:FETCh:SRWireless:PACKet:WLAN:CPOWer?

Function

Fetch cross power measurements.
Returns the power of each spatial stream of an S-MIMO or T-MIMO signal and indicates the dominant stream.

Currently no extension codes have been defined for this command. Use zero.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:CPOWer?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Dominant spatial stream number	Integer	---	---
5	Number of spatial streams	Integer	---	---
	For each spatial stream:	Integer	---	---
1	Average power	Real number	dBm	0.01 dB

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number

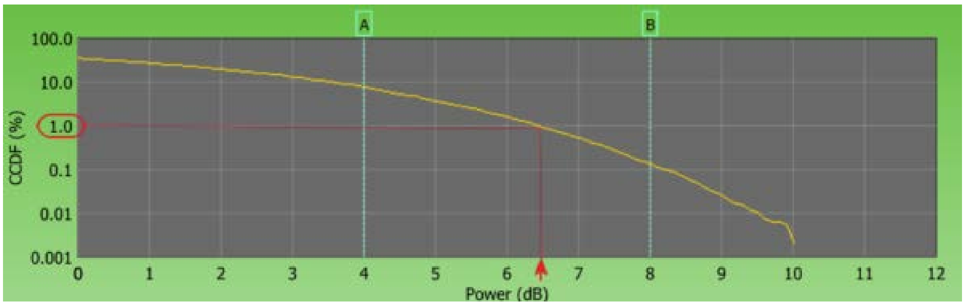
Example of Use

```
:FETC:SRW:PACK:WLAN:CPOW? 1,3,0
```

:FETCh:SRWireless:PACKet:WLAN:CCDF:PDISt?

Function

Fetch CCDF power distribution spot measurement, corresponding to the 'percentage of samples' value set using `:CONFigure:SRWireless:WLAN:CCDF:PSAMples`.



For the example CCDF measurement shown above, :CONFigure:SRWireless:WLAN:CCDF:PSAMples has been set to 1.0%. :FETC:SRW:SUMM:WLAN:CCDF:PDIS? will return the power distribution value corresponding to the point on the x-axis where the CCDF curve crosses the 1.0% line. In this case approximately 6.5 dB.

This means that the signal power exceeded the average power by at least 6.5 dB in 1% of the measured samples - (equivalent to 1% of the time)

Command

N/A

Query

`:FETCh:SRWireless:PACKet:WLAN:CCDF:PDISt?`

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code (reserved – always zero)	Integer	---	---
4	Percent samples	Real number	%	0.001%
5	Number of gates	Integer	---	---
	Results for first gate:			
1	Gate number	Integer	---	---
2	Power distribution	Real number	dB	0.01 dB
…	…			
	Results for last gate:			
1	Gate number	Integer	---	---
2	Power distribution	Real number	dB	0.01 dB

Parameters

#	Description	Format	Units	Min	Max	Def	Resolution
1	Segment index	Number	---	---	---	---	---
2	Packet index	Number	---	---	---	---	---
3	Extension code	Number	---	0	$2^{32}–1$	0	---

Example of Use

```
:FETC:SRW:PACK:WLAN:CCDF:PDIS? 1,1,0
```

:FETCh:SRWireless:PACKet:WLAN:DSSS:CONSt[:BINary]?

Function

Fetch constellation (graphical) measurements for a DSSS packet.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:DSSS:CONSt[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Number of points (IQ Pairs)	INT32	---
5	First I value	REAL32	---
6	First Q value	REAL32	---
…	…		
	Last I value	REAL32	---
	Last Q value	REAL32	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Note:

1. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Example of Use

```
:FETC:SRW:PACK:WLAN:DSSS:CONS? 1,4         // Fetch the constellation of
                                           // the 4th packet in segment 1.
```

:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM? (DEPRECATED)

Function

Fetch EVM (basic - numeric) measurements - DSSS packets.

Note:

This command is now deprecated. Please see the following commands, which clearly differentiate between results in percent and dB units:

```
:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:PCT?
:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:PCT?
:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:DB?
:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:DB?
```

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	DSSS EVM max - dB	Real number	dB	0.01 dB
4	DSSS EVM max - %	Real number	%	0.1 %
5	DSSS EVM min - dB	Real number	dB	0.01 dB
6	DSSS EVM min - %	Real number	%	0.1 %
7	DSSS EVM avg - dB	Real number	dB	0.01 dB
8	DSSS EVM avg %	Real number	%	0.1 0%
9	DSSS centre freq tolerance - ppm	Real number	ppm	0.1 ppm
10	DSSS centre freq tolerance Hz	Real number	Hz	1 Hz
11	DSSS phase error	Real number	degrees	0.01 degrees
12	DSSS magnitude error	Real number	---	0.01
13	DSSS IQ offset	Real number	---	0.01
14	DSSS chip clock error - ppm	Real number	ppm	0.1 ppm
15	DSSS chip clock error Hz	Real number	Hz	1 Hz
16	DSSS EVM test passed	Boolean	---	---
17	DSSS centre freq tolerance test passed	Boolean	---	---
18	DSSS chip clock test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:DSSS:EVM? 1,1
```

:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM? (Extended) (DEPRECATED)

Function

Fetch EVM (basic - numeric) measurements - DSSS packets – using extension code.

Note:

This command is now deprecated. Please see the following commands, which clearly differentiate between results in percent and dB units:

```
:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:PCT?
:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:PCT?
:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:DB?
:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:DB?
```

The optional extension code adds additional results to the response as described in the following table:

Extension code bit	Additional fields in response
0	DSSS carrier suppression from IQ offset
1	Amplitude and phase imbalance.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code	Integer	---	---
4	DSSS EVM (peak) max - dB	Real number	dB	0.01 dB
5	DSSS EVM (peak) max - %	Real number	%	0.1 %
6	DSSS EVM (peak) min - dB	Real number	dB	0.01 dB
7	DSSS EVM (peak) min - %	Real number	%	0.1 %
8	DSSS EVM (RMS) avg - dB	Real number	dB	0.01 dB
9	DSSS EVM (RMS) avg %	Real number	%	0.1 0%
10	DSSS centre freq tolerance - ppm	Real number	ppm	0.1 ppm
11	DSSS centre freq tolerance Hz	Real number	Hz	1 Hz
12	DSSS phase error	Real number	degrees	0.01 degrees
13	DSSS magnitude error	Real number	%	0.01
14	DSSS IQ offset	Real number	---	0.01
	If bit 0 of Extension Code is set:			
15	*DSSS carrier suppression from IQ offset*	*Real number*	*dB*	*0.01 dB*

16	DSSS chip clock error - ppm	Real number	ppm	0.1 ppm
17	DSSS chip clock error Hz	Real number	Hz	1 Hz
	If bit 1 of Extension Code is set:			
18	DSSS amplitude imbalance	Real number	dB	0.01 dB
19	DSSS phase imbalance	Real number	degrees	0.01 degrees
20	DSSS EVM test passed	Boolean	---	---
21	DSSS centre freq tolerance test passed	Boolean	---	---
22	DSSS chip clock test passed	Boolean	---	---
	If bit 0 of Extension Code is set:			
23	DSSS carrier suppression from IQ offset test passed.	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:DSSS:EVM? 1,1,1       //Extension code with bit 0 set.

> 1,1,1,-32.80,2.3,-73.70,0.0,-41.90,0.8,
-0.0,-0.0,0.33,0.57,-53.90,44.87,-0.0,-0,1,1,1,1
```

5 Remote Operation

:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:DB?

Function

Fetch WLAN EVM measurements in dB units.

This command replaces `:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM?` which has been marked as deprecated.

(See also `:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:PCT?`)

The extension code adds additional results to the response as described in the following table:

Extension code bit	Additional fields in response
0	DSSS carrier suppression from IQ offset
1	Amplitude and phase imbalance.

Notes:

The EVM limit check is performed on the EVM (peak) max measurement.

The IQ imbalance measurements (phase imbalance and amplitude imbalance) are only applicable to signals where the modulation type is QPSK. If there is no QPSK component in the signal to be measured, the values 0.0 degrees for phase imbalance and 0.0 dB for amplitude imbalance are returned.

IQ imbalance measurements are not applicable for the following cases:

1. Data rate of 1 Mbps. (Uses BPSK modulation only.)
2. Any DSSS packet type with a long preamble and long header (IEEE Long PPDU format) where the EVM calculation method is set to “PREAMBLE_HDR”.
   The reason is that selecting “PREAMBLE_HDR” limits the analysis to the preamble and header only, and for “long preamble and header” packets the modulation type is BPSK only.

Command

N/A

Query

`:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:DB?`

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code	Integer	---	---

4	DSSS EVM (RMS) max - dB	Real number	dB	0.01 dB
5	DSSS EVM (RMS) min - dB	Real number	dB	0.01 dB
6	DSSS EVM (RMS) avg dB	Real number	dB	0.01 dB
7	DSSS center freq tolerance - ppm	Real number	ppm	0.1 ppm
8	DSSS center freq tolerance Hz	Real number	Hz	1 Hz
9	DSSS phase error	Real number	degrees	0.01 degrees
10	DSSS magnitude error	Real number	%	0.01
11	DSSS IQ offset	Real number	---	0.01
	If bit 0 of Extension Code is set:			
12	DSSS carrier suppression from IQ offset	Real number	dB	0.01
13	DSSS chip clock error - ppm	Real number	ppm	0.1 ppm
14	DSSS chip clock error Hz	Real number	Hz	1 Hz
	If bit 1 of Extension Code is set:			
15	DSSS amplitude imbalance	Real number	dB	0.01 dB
16	DSSS phase imbalance	Real number	degrees	0.01 degrees
17	DSSS EVM test passed	Boolean	---	---
18	DSSS center freq tolerance test passed	Boolean	---	---
19	DSSS chip clock test passed	Boolean	---	---
	If bit 0 of Extension Code is set:			
20	DSSS carrier suppression from IQ offset test passed.	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:DSSS:EVM:DB? 1,1,1     // Bit 0 of extension code is set
```

5

Remote Operation

:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:PCT?

Function

Fetch WLAN EVM measurements in units of percent.

This command replaces `:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM?` which has been marked as deprecated.

(See also `:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:DB?`)

The extension code adds additional results to the response as described in the following table:

Extension code bit	Additional fields in response
0	DSSS carrier suppression from IQ offset
1	Amplitude and phase imbalance.

Notes:

The EVM limit check is performed on the EVM (peak) max measurement

The IQ imbalance measurements (phase imbalance and amplitude imbalance) are only applicable to signals where the modulation type is QPSK. If there is no QPSK component in the signal to be measured, the values 0.0 degrees for phase imbalance and 0.0 dB for amplitude imbalance are returned.

IQ imbalance measurements are not applicable for the following cases:

1. Data rate of 1 Mbps. (Uses BPSK modulation only.)
2. Any DSSS packet type with a long preamble and long header (IEEE Long PPDU format) where the EVM calculation method is set to “PREAMBLE_HDR”.
   The reason is that selecting “PREAMBLE_HDR” limits the analysis to the preamble and header only, and for “long preamble and header” packets the modulation type is BPSK only.

Command

N/A

Query

`:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:PCT?`

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code	Integer	---	---

4	DSSS EVM (peak) max - %	Real number	%	0.1 %
5	DSSS EVM (peak) min - %	Real number	%	0.1 %
6	DSSS EVM (RMS) avg %	Real number	%	0.1 %
7	DSSS center freq tolerance - ppm	Real number	ppm	0.1 ppm
8	DSSS center freq tolerance Hz	Real number	Hz	1 Hz
9	DSSS phase error	Real number	degrees	0.01 degrees
10	DSSS magnitude error	Real number	%	0.01
11	DSSS IQ offset	Real number	---	0.01
	If bit 0 of Extension Code is set:			
12	DSSS carrier suppression from IQ offset	Real number	dB	0.01
13	DSSS chip clock error - ppm	Real number	ppm	0.1 ppm
14	DSSS chip clock error Hz	Real number	Hz	1 Hz
	If bit 1 of Extension Code is set:			
15	DSSS amplitude imbalance	Real number	dB	0.01 dB
16	DSSS phase imbalance	Real number	degrees	0.01 degrees
17	DSSS EVM test passed	Boolean	---	---
18	DSSS center freq tolerance test passed	Boolean	---	---
19	DSSS chip clock test passed	Boolean	---	---
	If bit 0 of Extension Code is set:			
20	DSSS carrier suppression from IQ offset test passed.	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:DSSS:EVM:PCT? 1,1,1    // Bit 0 of extension code is set
```

5

Remote Operation

:FETCh:SRWireless:PACKet:WLAN:DSSS:PRAMps?

Function

Fetch Power profile (numeric) measurements - power up and down ramps – DSSS packets

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:DSSS:PRAMps?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Power-on ramp time	Real number	s	0.01 μs
4	Power-down ramp time	Real number	s	0.01 μs
5	Power-on ramp time test passed	Boolean	---	---
6	Power-down ramp time test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:DSSS:PRAM? 1,1
```

:FETCh:SRWireless:PACKet:WLAN:OFDM:AEVM?

Function

Fetch EVM (advanced - numeric) measurements for OFDM packets.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:OFDM:AEVM?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer		
2	Packet index	Integer		
3	Number of symbols	Integer		
4	Number of data subcarriers	Integer		
5	Number of pilot subcarriers	Integer		
6	Number of subcarriers			
7	First element of spectral flatness array	Real number	dB	0.01 dB
	…			
	Last element of spectral flatness array	Real number	dB	0.01 dB
	First element of EVM per data subcarrier array	Real number	dB	0.1 dB
	…			
	Last element of EVM per data subcarrier array	Real number	dB	0.1 dB
	First element of EVM per pilot subcarrier array	Real number	dB	0.1 dB
	…			
	Last element of EVM per pilot subcarrier array	Real number	dB	0.1 dB
	First element of EVM per symbol array	Real number	dB	0.1 dB
	…			
	Last element of EVM per symbol array	Real number	dB	0.1 dB

Notes:

1. Length of spectral flatness array is equal to the number of subcarriers (field #6).
2. Length of EVM per data subcarrier array is equal to the number of data subcarriers (field #4).
3. Length of EVM per pilot subcarrier array is equal to the number of pilot subcarriers (field #5).
4. Length of EVM per symbol array is equal to the number of symbols (field #3).

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:OFDM:AEVM? 1,2
```

5 Remote Operation

:FETCh:SRWireless:PACKet:WLAN:OFDM:CESTimation?

Function

Fetch the channel estimation values (amplitude and phase).

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:OFDM:CESTimation?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Number of spatial streams	Integer	---	---
5	Number of values each of amplitude and phase results	Integer	---	---
6	First amplitude value	Real number	V / Hz	0.000001
	…	---	---	---
	Last amplitude value	Real number	V / Hz	0.000001
	First phase value	Real number	radians	0.000001
	…	---	---	---
	Last phase value	Real number	radians	0.000001

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number (currently zero)

Example of Use

```
:FETC:SRW:PACK:WLAN:OFDM:CEST? 1,5,0
```

:FETCh:SRWireless:PACKet:WLAN:OFDM:CESTimation:AMPLitude?

Function

Fetch the channel estimation values (amplitude only).

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:OFDM:CESTimation:AMPLitude?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Number of spatial streams	Integer	---	---
5	Number of amplitude values	Integer	---	---
6	First amplitude value	Real number	V / Hz	0.000001
	...	---	---	---
	Last amplitude value	Real number	V / Hz	0.000001

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number (currently zero)

Example of Use

```
:FETC:SRW:PACK:WLAN:OFDM:CEST:AMPL? 1,5,0
```

:FETCh:SRWireless:PACKet:WLAN:OFDM:CESTimation:PHASe?

Function

Fetch the channel estimation values (phase only).

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:OFDM:CESTimation:PHASe?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Number of spatial streams	Integer	---	---
5	Number of phase values	Integer	---	---
6	First phase value	Real number	radians	0.000001
	…	---	---	---
	Last phase value	Real number	radians	0.000001

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number (currently zero)

Example of Use

```
:FETC:SRW:PACK:WLAN:OFDM:CEST:PHAS? 1,5,0
```

:FETCh:SRWireless:PACKet:WLAN:OFDM:CONSt[:BINary]?

Function

Fetch constellation (graphical) measurements for an OFDM packet.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:OFDM:CONSt[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Number of points (IQ Pairs)	INT32	---
5	First I value	REAL32	---
6	First Q value	REAL32	---
…	…		
	Last I value	REAL32	---
	Last Q value	REAL32	---
	Index of start of pilot IQ data in IQ array (IQ Pairs).	INT32	---

Note that the pilot IQ data is located towards the end of the array and its position is indicated by the 'index of start of pilot IQ data' field. This allows the pilot data to be removed from a plot of the constellation or for it to be plotted in a different color.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Note:

1. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Example of Use

```
:FETC:SRW:PACK:WLAN:OFDM:CONS? 1,3        // Fetch the constellation of
                                          // the third packet in segment 1.
```

:FETCh:SRWireless:PACKet:WLAN:OFDM:EVM?

Function

Fetch EVM (basic - numeric) measurements - OFDM packets.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:OFDM:EVM?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer		
2	Packet index	Integer		
3	OFDM number of subcarriers	Integer		
4	OFDM number of symbols	Integer		
5	OFDM centre frequency leakage	Real number	dB	0.01 dB
6	OFDM symbol clk freq tolerance ppm	Real number	ppm	0.1 ppm
7	OFDM centre freq tolerance ppm	Real number	ppm	0.1 ppm
8	OFDM centre freq tolerance Hz	Real number	Hz	1 Hz
9	OFDM EVM max - dB	Real number	dB	0.01 dB
10	OFDM EVM max - %	Real number	%	0.01 %
11	OFDM EVM min - dB	Real number	dB	0.01 dB
12	OFDM EVM min - %	Real number	%	0.01 %
13	OFDM EVM avg - dB	Real number	dB	0.01 dB
14	OFDM EVM avg %	Real number	%	0.01 %
15	OFDM centre frequency leakage test passed	Boolean	---	---
16	OFDM symbol clk freq tolerance test passed	Boolean	---	---
17	OFDM centre freq tolerance test passed	Boolean	---	---
18	OFDM EVM test passed	Boolean	---	---
19	OFDM spectral flatness test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:OFDM:EVM? 1,1
```

:FETCh:SRWireless:PACKet:WLAN:OFDM:EVM? (Extended)

Function

Fetch EVM (basic - numeric) measurements - OFDM packets.

The optional extension code adds additional results to the response as described in the following table:

Extension code bit	Additional fields in response
0	IQ imbalance and phase error measurements.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:OFDM:EVM?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer		
2	Packet index	Integer		
3	Extension code	Integer		
4	OFDM number of subcarriers	Integer		
5	OFDM number of symbols	Integer		
6	OFDM centre frequency leakage	Real number	dB	0.01 dB
7	OFDM symbol clk freq tolerance ppm	Real number	ppm	0.1 ppm
8	OFDM centre freq tolerance ppm	Real number	ppm	0.1 ppm
9	OFDM centre freq tolerance Hz	Real number	Hz	1 Hz
10	OFDM EVM max - dB	Real number	dB	0.01 dB
11	OFDM EVM max - %	Real number	%	0.01 %
12	OFDM EVM min - dB	Real number	dB	0.01 dB
13	OFDM EVM min - %	Real number	%	0.01 %
14	OFDM EVM avg - dB	Real number	dB	0.01 dB
15	OFDM EVM avg %	Real number	%	0.01 %
	If bit 0 of extension code set:			
16	OFDM amplitude imbalance	Real number	dB	0.01 dB
17	OFDM phase imbalance	Real number	degrees	0.01°
18	OFDM phase error	Real number	degrees	0.01°
19	OFDM centre frequency leakage test passed	Boolean	---	---
20	OFDM symbol clk freq tolerance test passed	Boolean	---	---

21	OFDM centre freq tolerance test passed	Boolean	---	---
22	OFDM EVM test passed	Boolean	---	---
23	OFDM spectral flatness test passed	Boolean	---	---

Note:

Amplitude imbalance, phase imbalance and phase error are not available when the wireless standard defined for the segment is composite MIMO (COMPMIMO). Not a number (NaN) values are returned for these measurements in that case.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:OFDM:EVM? 1,1,1
```

:FETCh:SRWireless:PACKet:WLAN:OFDM:EVM:COMBined?

Function

Fetch combined WLAN EVM measurements.
Combined EVM measurements are available for 802.11ac 80+80 signals, where a combined EVM of the upper and lower frequency segments is calculated.

There are currently no extension codes defined for this command.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:OFDM:EVM:COMBined?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	OFDM combined EVM max – dB (the maximum EVM measured in all related capture segments)	Real number	dB	0.1 dB
5	OFDM combined EVM max – % (the maximum EVM measured in all related capture segments)	Real number	%	0.1%
6	OFDM combined EVM min – dB (the minimum EVM measured in all related capture segments)	Real number	dB	0.1 dB
7	OFDM combined EVM min – % (the minimum EVM measured in all related capture segments)	Real number	%	0.1%
8	OFDM combined EVM avg – dB (the average EVM measured in all related capture segments)	Real number	dB	0.1 dB
9	OFDM combined EVM avg – % (the average EVM measured in all related capture segments)	Real number	%	0.1%
10	OFDM combined EVM test passed	Boolean		

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:OFDM:EVM:COMB? 1,1,0
```

5 Remote Operation

:FETCh:SRWireless:PACKet:WLAN:OFDM:FERRor?

Function

Fetch OFDM preamble frequency error v time measurement.

Currently no extension codes have been defined for this command. Use zero.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:OFDM:FERRor?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Timebase length	Integer	μs	1 μs
5	Samples / μs	Integer	---	---
6	Center frequency tolerance	Integer	Hz	1 Hz
7	Number of sample periods in timebase length	Integer	---	---
8	Offset (in samples) to start of short training sequence	Integer	---	---
9	Number of samples in short training sequence	Integer	---	---
	For each sample in short training sequence:	Integer	---	---
	Frequency error	Integer	Hz	1 Hz
	Offset (in samples) to start of long training sequence	Integer	---	---
	Number of samples in long training sequence	Integer	---	---
	For each sample in long training sequence:	Integer	---	---
	Frequency error	Integer	Hz	1 Hz

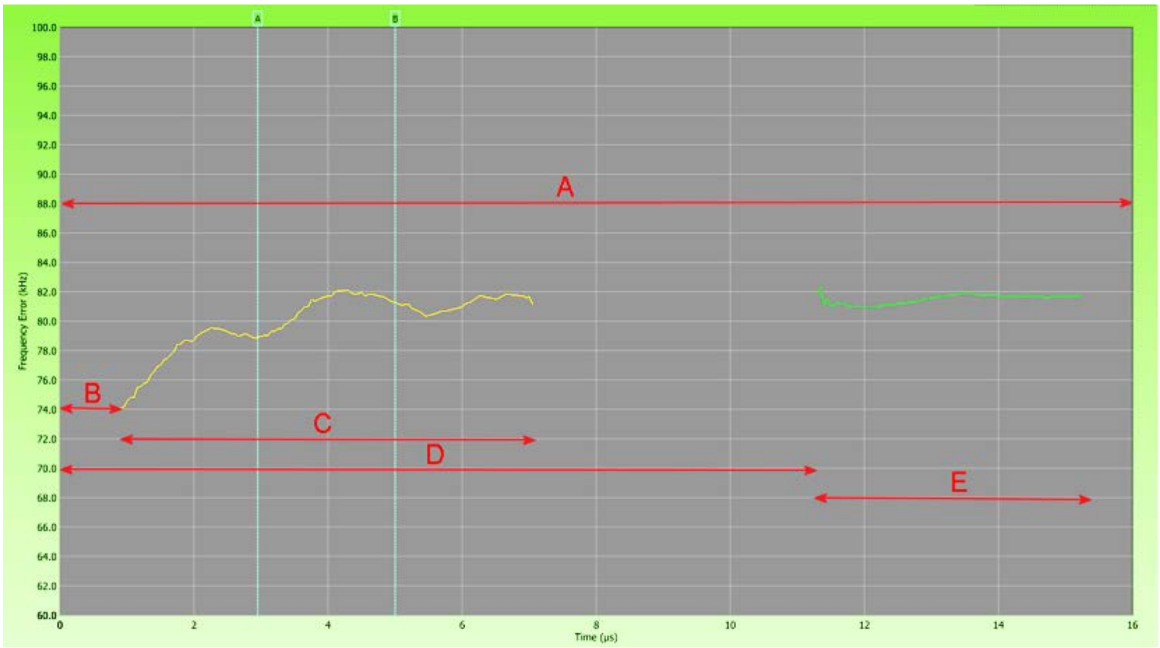
Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:OFDM:FERR? 1,7,0
```

Using the Response Fields to Display the Measurement



The graph shows a display of frequency error v time during a packet preamble. The yellow trace shows the short training sequence and the green trace shows the long training sequence plotted on a timebase spanning the packet preamble.

The following table explains how this graph can be constructed from the **response fields** returned by this command.

Timebase Length	See **A**. This is the length of the OFDM packet preamble in microseconds. Within this time frame are plotted the frequency error graphs for two fields in the preamble: the short training sequence and the long training sequence.
Number of sample periods in Timebase Length	This is the total number of points that can be plotted within the Timebase Length. (It is equal to **Timebase Length** multiplied by **Samples / µs**.)
Offset (in samples) to start of short training sequence	See **B**. This value defines where along the x axis to start plotting the short training sequence results.
Number of samples in short training sequence	See **C**. This value defines the length (in samples) of the short training sequence graph.
Offset (in samples) to start of short training sequence	See **D**. This value defines where along the x axis to start plotting the long training sequence results.
Number of samples in long training sequence	See **E**. This value defines the length (in samples) of the long training sequence graph.

:FETCh:SRWireless:PACKet:WLAN:OFDM:PERRor?

Function

Fetch OFDM phase error v symbol measurement.

Currently no extension codes have been defined for this command. Use zero.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:OFDM:PERRor?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Number of symbols	Integer	---	---
	For each symbol:			
1	Phase error	Real number	degrees	0.01°

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:OFDM:PERR? 1,3,0
```

:FETCh:SRWireless:PACKet:WLAN:POWer?

Function

Fetch WLAN transmit power (numeric) measurements.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:POWer?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Number of gates	Integer	---	---
	Results for first gate:			
1	Gate number	Integer	---	---
2	Peak power	Real number	dBm	0.01 dB
3	Average power	Real number	dBm	0.01 dB
4	Crest factor	Real number	dB	0.01 dB
5	Average power upper limit test passed	Boolean	---	---
6	Average power lower limit test passed	Boolean	---	---
…	...			
	Results for last gate:			
1	Gate number	Integer	---	
2	Peak power	Real number	dBm	0.01 dB
3	Average power	Real number	dBm	0.01 dB
4	Crest factor	Real number	dB	0.01 dB
5	Average power upper limit test passed	Boolean	---	---
6	Average power lower limit test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:POW? 1,5
```

:FETCh:SRWireless:PACKet:WLAN:PPRofile:AVERage:ASCii?

Function

Fetch WLAN power profile (graphical) measurements - average - ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:PPRofile:AVERage:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Start time	Real number	s	1 μs
4	Stop time	Real number	s	1 μs
5	Number of points in power profile	Integer	---	---
6	First power value	Real number	dBm	0.01 dB
	…			
	Last power value	Real number	dBm	0.01 dB

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:PPR:AVER:ASC? 1,2
```

:FETCh:SRWireless:PACKet:WLAN:PPRofile:AVERage[:BINary]?

Function

Fetch WLAN power profile (graphical) measurements - binary format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:PPRofile:AVERage[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary Header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Start time in seconds	REAL32	s
5	Stop time in seconds	REAL32	s
6	Number of points	INT32	---
7	Power in dBm	REAL32	dBm

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Notes:

1. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
2. Binary data will always be a multiple of 4 bytes.
3. Refer to sections 5.3.3 and 5.3.4 *Binary_Header_Information* and *Binary_Data_Fields*.

Example of Use

```
:FETC:SRW:PACK:WLAN:PPR:AVER? 1,3
```

:FETCh:SRWireless:PACKet:WLAN:PPRofile:PEAK:ASCii?

Function

Fetch WLAN power profile (graphical) measurements - peak - ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:PPRofile:PEAK:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Start time	Real number	s	1 μs
4	Stop time	Real number	s	1 μs
5	Number of points in power profile	Integer	---	---
6	First power value	Real number	dBm	0.01 dB
	…			
	Last power value	Real number	dBm	0.01 dB

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:PPR:PEAK:ASC? 1,3
```

:FETCh:SRWireless:PACKet:WLAN:PPRofile:PEAK[:BINary]?

Function

Fetch WLAN power profile (graphical) measurements - peak - binary format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:PPRofile:PEAK[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary Header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Start time in seconds	REAL32	s
5	Stop time in seconds	REAL32	s
6	Number of points	INT32	---
7	Power in dBm	REAL32	dBm

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:PPR:PEAK? 1,2
```

Notes:

1. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
2. Binary data will always be a multiple of 4 bytes.
3. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

:FETCh:SRWireless:PACKet:WLAN:PRAMps?

Function

Fetch WLAN power profile (numeric) measurements - power up and down ramps – WLAN packets

Note:

This command supersedes `:FETCh:SRWireless:PACKet:WLAN:DSSS:PRAMps` following a change to support power ramp measurements on OFDM as well as DSSS signals. The use of the older command is now deprecated.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:PRAMps?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Power-on ramp time for packet	Real number	s	0.01 µs
4	Power-down ramp time for packet	Real number	s	0.01 µs
5	Power-on ramp time test passed	Boolean	---	---
6	Power-down ramp time test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:PRAM? 1,2
```

:FETCh:SRWireless:PACKet:WLAN:SPECtrum:GRAPh:ASCii?

Function

Fetch WLAN transmitter spectrum (graphical) measurements - ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:SPECtrum:GRAPh:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Number of gates	Integer	---	---
	For first gate:			
1	Gate number	Integer	---	---
2	Bandwidth	Real number	Hz	1 Hz
3	Number of points in spectrum	Integer	---	---
4	First power value	Real number	dBm	0.01 dB
	…			
	Last power value	Real number	dBm	0.01 dB
	…			
	For last gate:			
1	Gate number	Integer	---	---
2	Bandwidth	Real number	Hz	1 Hz
3	Number of points in spectrum	Integer	---	---
4	First power value	Real number	dBm	0.01 dB
	…			
	Last power value	Real number	dBm	0.01 dB

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:SPEC:GRAP:ASC? 1,2
```

5

Remote Operation

:FETCh:SRWireless:PACKet:WLAN:SPECtrum:GRAPh[:BINary]?

Function

Fetch WLAN transmitter spectrum (graphical) measurements - binary format.

Command

N/A

Query

`:FETCh:SRWireless:PACKet:WLAN:SPECtrum:GRAPh[:BINary]?`

Response

#	Description	Format	Units
1	#nmm…mm	Binary Header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Number of gates	INT32	---
	For first gate:		
1	Gate number	INT32	---
2	Bandwidth	REAL32	Hz
3	Number of points	INT32	---
4	Power in dBm	REAL32	dBm
…	…		
n	Last power value	REAL32	dBm
…	…		
	For last gate:		
1	Gate number	INT32	---
2	Bandwidth	REAL32	Hz
3	Number of points	INT32	---
4	Power in dBm	REAL32	dBm
…	…		
n	Last power value	REAL32	dBm

Note:

The bandwidth fields can take the value (10, 70, 130 or 160 MHz, depending on the wireless standard. The x-axis for the graph data therefore runs from –BW/2 to +BW/2.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:SPEC:GRAP? 1,7
```

Notes:

1. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
2. Binary data will always be a multiple of 4 bytes.
3. Refer to sections 5.3.3 and 5.3.4 *Binary_Header_Information* and *Binary_Data_Fields*.

:FETCh:SRWireless:PACKet:WLAN:SPECtrum:GRAPh:COMBined:ASCii?

Function

Fetch WLAN transmitter spectrum (graphical) combined 802.11ac 80+80 measurements - ASCII format.

Command

N/A

Query

`:FETCh:SRWireless:PACKet:WLAN:SPECtrum:GRAPh:COMBined:ASCii?`

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Number of gates	Integer	---	---
	For each gate:			
5	Gate number	Integer	---	---
6	Number of spectral profiles (See note)	Integer	---	---
	For each spectral profile			
7	Spectral profile number (1 or 2)	Integer		
8	Carrier frequency / carrier offset (See note)	Integer	Hz	1 Hz
9	Bandwidth	Integer	Hz	1 Hz
10	Number of points	Integer	---	---
	For each point			
11	Power value	Real number	dBm	0.01 dB

Notes:

The **number of spectral profiles** can be 1 or 2. If it is 1 it means that the two carriers forming the 80+80 transmission are less than or equal to 280 MHz apart. If it is 2 it means that the carriers are more than 280 MHz apart.

The **carrier frequency / carrier offset** field contains different information depending on the separation of the carriers.
If the carriers are less than or equal to 280 MHz apart, this value is the offset in Hz between the carrier frequency and the center of the profile. The center of the profile is defined as 0 Hz.
If the carriers are more than 280 MHz apart, this value is the frequency of the carrier.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code (currently zero)	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:SPEC:GRAP:COMB:ASC? 1,2,0
```

:FETCh:SRWireless:PACKet:WLAN:SPECtrum:GRAPh:COMBined[:BINary]?

Function

Fetch WLAN transmitter spectrum (graphical) combined 802.11ac 80+80 measurements - binary format.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:SPECtrum:GRAPh:COMBined[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary Header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Extension code (currently zero)	INT32	---
5	Number of gates	INT32	---
	For each gate:		
6	Gate number	INT32	---
7	Number of spectral profiles (See notes)	INT32	---
	For each spectral profile	INT32	---
8	Spectral profile number (1 or 2)	INT32	---
9	Carrier frequency / carrier offset (See notes)	INT32	Hz
10	Bandwidth	INT32	Hz
11	Number of points	INT32	---
	For each point		
12	Power value	REAL32	dBm

Notes:

The **number of spectral profiles** can be 1 or 2. If it is 1 it means that the two carriers forming the 80+80 transmission are less than or equal to 280 MHz apart. If it is 2 it means that the carriers are more than 280 MHz apart.

The **carrier frequency / carrier offset** field contains different information depending on the separation of the carriers.
If the carriers are less than or equal to 280 MHz apart, this value is the offset in Hz between the carrier frequency and the center of the profile. The center of the profile is defined as 0 Hz.
If the carriers are more than 280 MHz apart, this value is the frequency of the carrier.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code (currently zero)	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:SPEC:GRAP:COMB? 1,7,0
```

Notes:

1. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
2. Binary data will always be a multiple of 4 bytes.
3. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

:FETCh:SRWireless:PACKet:WLAN:SPECtrum:MASK:COMBined:ASCii?

Function

Fetch WLAN transmitter spectrum mask for 802.11ac 80+80 measurements - ASCII format. The shape of the mask depends on the separation of the 80+80 carriers and the relative power level of the two signals.

Command

N/A

Query

`:FETCh:SRWireless:PACKet:WLAN:SPECtrum:MASK:COMBined:ASCii?`

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Gate number	Integer	---	---
5	Number of spectral profiles (See note)	Integer	---	---
	For each spectral profile			
6	Spectral profile number (1 or 2)	Integer	---	---
7	Carrier frequency / carrier offset (See note)	Integer	Hz	1 Hz
8	Bandwidth	Integer	Hz	1 Hz
9	Number of points	Integer	---	---
	For each point			
10	Mask power value	Real number	dBm	0.01 dB

Notes:

The **number of spectral profiles** can be 1 or 2. If it is 1 it means that the two carriers forming the 80+80 transmission are less than or equal to 280 MHz apart. If it is 2 it means that the carriers are more than 280 MHz apart.

The **carrier frequency / carrier offset** field contains different information depending on the separation of the carriers.
If the carriers are less than or equal to 280 MHz apart, this value is the offset in Hz between the carrier frequency and the center of the profile. The center of the profile is defined as 0 Hz.
If the carriers are more than 280 MHz apart, this value is the frequency of the carrier.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code (currently zero)	Number
4	Gate number	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:SPEC:MASK:COMB:ASC? 1,2,0,1
```

:FETCh:SRWireless:PACKet:WLAN:SPECtrum:MASK:COMBined[:BINary]?

Function

Fetch WLAN transmitter spectrum mask for 802.11ac 80+80 measurements - binary format. The shape of the mask depends on the separation of the 80+80 carriers and the relative power level of the two signals.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:SPECtrum:MASK:COMBined[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary Header	---
2	Segment index	INT32	---
3	Packet index	INT32	---
4	Extension code (currently zero)	INT32	---
5	Gate number	INT32	---
6	Number of spectral profiles (See notes)	INT32	---
	For each spectral profile	INT32	---
7	Spectral profile number (1 or 2)	INT32	---
8	Carrier frequency / carrier offset (See notes)	INT32	Hz
9	Bandwidth	INT32	Hz
10	Number of points	INT32	---
	For each point		
11	Mask power value	REAL32	dBm

Notes:

The **number of spectral profiles** can be 1 or 2. If it is 1 it means that the two carriers forming the 80+80 transmission are less than or equal to 280 MHz apart. If it is 2 it means that the carriers are more than 280 MHz apart.

The **carrier frequency / carrier offset** field contains different information depending on the separation of the carriers.
If the carriers are less than or equal to 280 MHz apart, this value is the offset in Hz between the carrier frequency and the center of the profile. The center of the profile is defined as 0 Hz.
If the carriers are more than 280 MHz apart, this value is the frequency of the carrier.

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code (currently zero)	Number
4	Gate number	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:SPEC:MASK:COMB? 1,7,0,1
```

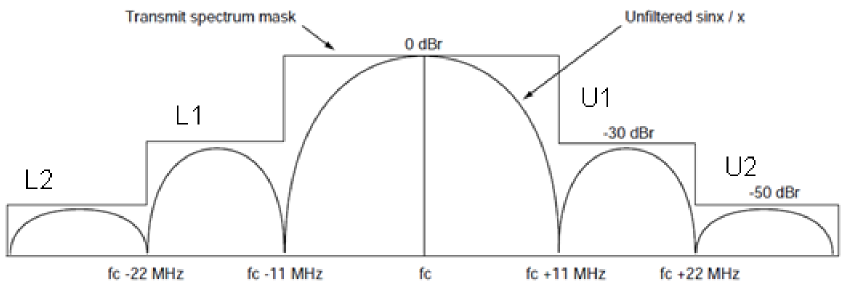
Notes:

1. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
2. Binary data will always be a multiple of 4 bytes.
3. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

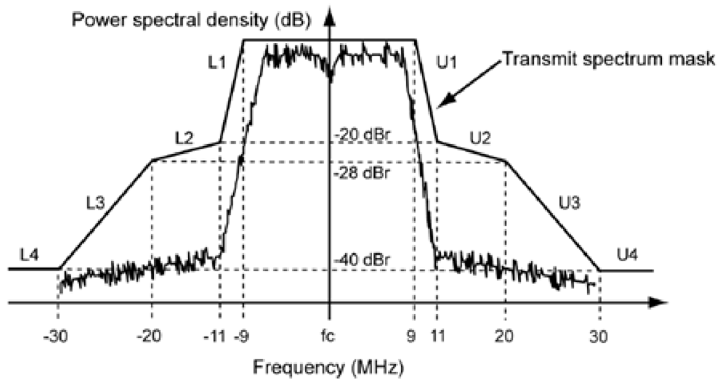
:FETCh:SRWireless:PACKet:WLAN:SPECtrum:NUMeric?

Function

Fetch WLAN transmitter spectrum (numeric) measurements.
Limit masks are divided into segments. Segments below the carrier have the designation 'L' for 'Lower'; and those above the carrier are designated 'U' for 'Upper'. DSSS masks have two segments above and below the carrier: L2, L1, U1 and U2.



OFDM masks have four segments above and below the carrier: L4 to L1 and U1 to U4.



Measurements are returned for each segment of the limit mask as follows:

- The frequency offset in Hz from the carrier to the highest signal peak within the mask segment.
- The power of the signal peak relative to the limit mask at that frequency offset.
- The absolute power of the signal peak at that frequency offset.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:SPECtrum:NUMeric?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Number of gates	Integer	---	---
	Results for each gate:			
1	Gate number	Integer	---	---
2	RF carrier suppression (DSSS)	Real number	dB	0.01 dB
3	Power spectral density	Real number	dBm / MHz	0.01 dB
4	Occupied bandwidth	Real number	Hz	1 Hz
5	Spectrum mask passed limit test	Boolean	---	---
6	RF carrier suppression (DSSS) passed	Boolean	---	---
7	Power spectral density test passed	Boolean	---	---
8	Occupied bandwidth test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:SPEC:NUM? 1,6
```

:FETCh:SRWireless:PACKet:WLAN:SPECtrum:NUMeric? (Extended)

Function

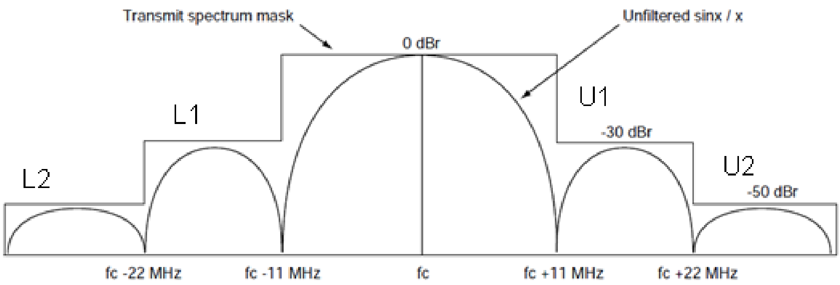
Fetch WLAN transmitter spectrum (numeric) measurements with optional extension code.

Extension code bit	Additional fields in response
0	Mask margin results.
1	Mask violation percentage.
2	Power spectral density in units of mW / MHz.
3	Extends the mask margin results (selected by bit 0 to include results for the additional mask segment applicable to 802.11p regulatory masks associated with on of the four power classes (A to D). This bit should only be set when measuring an 802.11p signal against the specification defined by the power class.

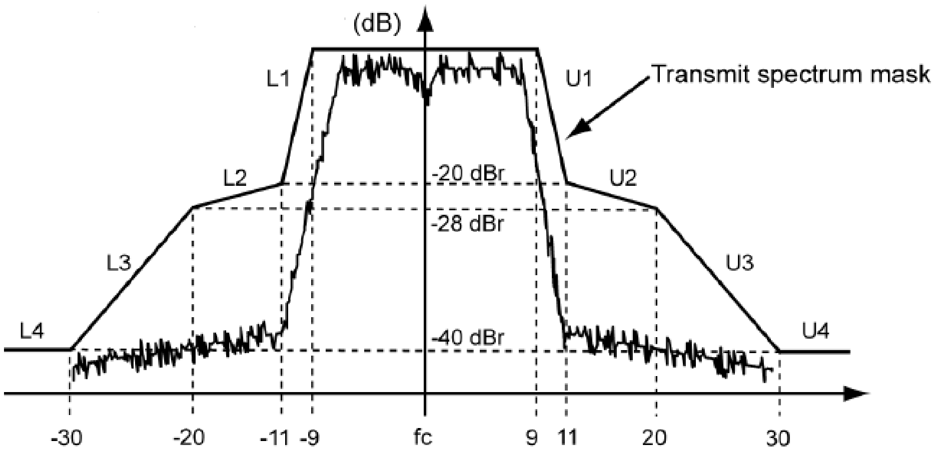
These measurement options are described in detail below:

Mask Margin Results

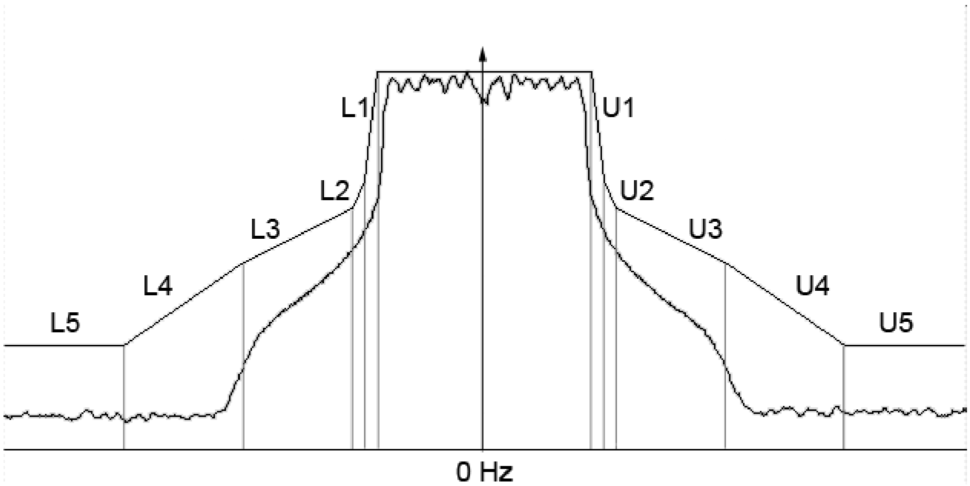
If bit 0 of the extension code is set, the response includes mask margin results for the packet. Limit masks are divided into segments. Segments below the carrier have the designation ‘L’ for ‘Lower’; and those above the carrier are designated ‘U’ for ‘Upper’. DSSS masks defined in the IEEE 802.11 standard have two segments above and below the carrier: L2, L1, U1 and U2.



OFDM masks defined in the IEEE 802.11 standard have four segments above and below the carrier: L4 to L1 and U1 to U4.



802.11p regulatory masks associated with one of the four power classes: A, B, C or D have an additional pair of segments as shown in the figure and table below.



	Frequency offset from carrier (MHz)									
	L5	**L4**	**L3**	**L2**	**L1**	**U1**	**U2**	**U3**	**U4**	**U5**
5 MHz	–7.5	–5	–2.75	–2.5	–2.25	+2.25	+2.5	+2.75	+5	+7.5
10 MHz	–15	–10	–5.5	–5	–4.5	+4.5	+5	+5.5	+10	+15
20 MHz	–30	–20	–11	–10	–9	+9	+10	+11	+20	+30

The table above shows the frequency offsets from the carrier of the start of each segment of the 802.11p spectral mask for each signal bandwidth. Note that there are 10 segments (L5 to U5) compared with the IEEE standard mask, which has 8 segments. Remember to set bit 3 of the extension code when measuring 802.11p signals against the regulatory masks. This will ensure that results for all 10 segments appear in the response.

Measurements are returned for each segment of the limit mask as follows:

- The frequency offset in Hz from the carrier to the highest signal peak within the mask segment.
- The power of the signal peak relative to the limit mask at that frequency offset.
- The absolute power of the signal peak at that frequency offset.

Mask Violation Percentage
If bit 1 of the extension code is set, the instrument returns the mask violation percentage value for the spectrum measurement. Mask violation is the proportion of the trace that exceeds the mask limit. It is used to indicate the 'degree of failure'. If the mask is crossed at only a few frequency points, the violation percentage value will be small. A more serious breach of the limit will result in a higher violation percentage value.

Power Spectral Density (mW / MHz)
If bit 2 of the extension code is set the instrument returns the power spectral density result in units of mW / MHz as well as the equivalent result in dBm / MHz, which is always returned in the response

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:SPECtrum:NUMeric?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code	Integer	---	---
4	Number of gates	Integer	---	---
	Results for each gate:			
1	Gate number	Integer	---	---
2	RF carrier suppression (DSSS)	Real number	dB	0.01 dB
3	Power spectral density	Real number	dBm / MHz	0.01 dB
	If bit 2 of extension code set:			
4	Power spectral density (mW)	Real number	mW / MHz	3 sig figs
5	Occupied bandwidth	Real number	Hz	1 Hz
	If bit 0 of extension code set:			
	If bit 3 of extension code set (as well as bit 0)			
6	L5 - frequency offset from carrier	Real number	Hz	1 Hz
7	L5 - power relative to mask	Real number	dB	0.01 dB
8	L5 - absolute power	Real number	dBm	0.01 dB

9	L4 - frequency offset from carrier	Real number	Hz	1 Hz
10	L4 - power relative to mask	Real number	dB	0.01 dB
11	L4 - absolute power	Real number	dBm	0.01 dB
12	L3 - frequency offset from carrier	Real number	Hz	1 Hz
13	L3 - power relative to mask	Real number	dB	0.01 dB
14	L3 - absolute power	Real number	dBm	0.01 dB
15	L2 - frequency offset from carrier	Real number	Hz	1 Hz
16	L2 - power relative to mask	Real number	dB	0.01 dB
17	L2 - absolute power	Real number	dBm	0.01 dB
18	L1 - frequency offset from carrier	Real number	Hz	1 Hz
19	L1 - power relative to mask	Real number	dB	0.01 dB
20	L1 - absolute power	Real number	dBm	0.01 dB
21	U1 - frequency offset from carrier	Real number	Hz	1 Hz
22	U1 - power relative to mask	Real number	dB	0.01 dB
23	U1 - absolute power	Real number	dBm	0.01 dB
24	U2 - frequency offset from carrier	Real number	Hz	1 Hz
25	U2 - power relative to mask	Real number	dB	0.01 dB
26	U2 - absolute power	Real number	dBm	0.01 dB
27	U3 - frequency offset from carrier	Real number	Hz	1 Hz
28	U3 - power relative to mask	Real number	dB	0.01 dB
29	U3 - absolute power	Real number	dBm	0.01 dB
30	U4 - frequency offset from carrier	Real number	Hz	1 Hz
31	U4 - power relative to mask	Real number	dB	0.01 dB
32	U4 - absolute power	Real number	dBm	0.01 dB
	If bit 3 of extension code set (as well as bit 0)			
33	U5 - frequency offset from carrier	Real number	Hz	1 Hz
34	U5 - power relative to mask	Real number	dB	0.01 dB
35	U5 - absolute power	Real number	dBm	0.01 dB

	..If bit 1 of extension code set:			
36	..Mask violation	Real number	%	0.01 %

	The following fields are always present:			
37	Spectrum mask passed limit test	Boolean	---	---
38	RF carrier suppression (DSSS) passed	Boolean	---	---
39	Power spectral density test passed	Boolean	---	---
40	Occupied bandwidth test passed	Boolean	---	---

5 Remote Operation

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:SPEC:NUM? 1,1,15      // Bits 0,1,2 and 3 of extension code are
                                          // set. All results will be returned.
```

:FETCh:SRWireless:PACKet:WLAN:SPECtrum:SPOT?

Function

Fetch spot spectrum measurements at a frequency offset above and below the carrier frequency (f_c).

Note that the specified frequency must be one of those previously defined using :CONFigure:SRWireless:WLAN:SPOT:FREQuency:ADD.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:SPECtrum:SPOT?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index	Integer	---	---
3	Extension code (reserved – always zero)	Integer	---	---
4	Number of gates	Integer	---	---
	Results for first gate:			
1	Gate number	Integer	---	---
2	Frequency offset	Real number	Hz	1 Hz
3	Absolute power at f_c + offset	Real number	dBm	0.01 dB
4	Relative power compared to limit mask at f_c + offset	Real number	dB	0.01 dB
5	Relative power compared to signal peak at f_c + offset	Real number	dB	0.01 dB
6	Absolute power at f_c – offset	Real number	dBm	0.01 dB
7	Relative power compared to limit mask at f_c – offset	Real number	dB	0.01 dB
8	Relative power compared to signal peak at f_c – offset	Real number	dB	0.01 dB
…	…			
	Results for last gate:			
1	Gate number	Integer	---	---
2	Frequency offset	Real number	Hz	1 Hz
3	Absolute power at fc + offset	Real number	dBm	0.01 dB
4	Relative power compared to limit mask at fc + offset	Real number	dB	0.01 dB
5	Relative power compared to signal peak at fc + offset	Real number	dB	0.01 dB
6	Absolute power at fc – offset	Real number	dBm	0.01 dB
7	Relative power compared to limit mask at fc – offset	Real number	dB	0.01 dB
8	Relative power compared to signal peak at fc – offset	Real number	dB	0.01 dB

Parameters

#	Description	Format	Units	Min	Max	Def	Resolution
1	Segment index	Number	---	---	---	---	---
2	Packet index	Number	---	---	---	---	---
3	Extension code	Number	---	0	$2^{32}–1$	0	---
4	Frequency offset	Number	Hz	0	See note	0	1 Hz

Note:
The maximum frequency offset is equal to half the spectrum bandwidth. The spectrum bandwidth depends on the wireless standard.

Example of Use

```
:FETC:SRW:PACK:WLAN:SPEC:SPOT? 1,1,0,20E+06      // Extract spot readings
                                                  // at ±20 MHz.
> 1,1,0,1,1,20000000,
  -48.04,-12.12,-38.35,                           // fc + 20 MHz
  -47.83,-11.91,-38.49                            // fc – 20 MHz
```

:FETCh:SRWireless:PACKet:WLAN:SPOWer?

Function

Fetch the composite MIMO stream power and related measurements.

Command

N/A

Query

```
:FETCh:SRWireless:PACKet:WLAN:SPOWer?
```

Response

#	Description	Format	Units
1	Segment index	Integer	---
2	Packet index	Integer	---
3	Extension code	Integer	---
4	Number of spatial streams	Integer	---
	For first spatial stream:		
1	Average stream power	Real number	dBm
2	Peak stream power	Real number	dBm
3	Stream crest factor	Real number	dB
…			
	For last spatial stream:		
1	Average stream power	Real number	dBm
2	Peak stream power	Real number	dBm
3	Stream crest factor	Real number	dB

Parameters

#	Description	Format
1	Segment index	Number
2	Packet index	Number
3	Extension code	Number

Example of Use

```
:FETC:SRW:PACK:WLAN:SPOW? 1,2,0
```

5

Remote Operation

:FETCh:SRWireless:SEGMent:IDENtity?

Function

Fetch information about packets that have been captured and identified automatically. Following capture of a segment where the wireless standard is AUTODSSS, AUTOOFDM or AUTOBT, this command may be used to read information about the type of packets that the system has identified.

Note:

If the instrument receives a mixture of packets of different standards, the wireless standard identified by this command will be that of the **first** packet received that conforms to a supported IEEE standard. Only packets that match this standard will be included in summary measurements.

Command

N/A

Query

`:FETCh:SRWireless:SEGMent:IDENtity?`

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code	Integer	---	---
4	Segment status	Integer	---	---
5	Decoded standard	See definition	---	---
	If decoded standard is Bluetooth / BLE			
6	Bluetooth address (LAP)	Hexadecimal	---	---
7	Bluetooth packet type	See definition	---	---
8	Bluetooth payload type	See definition	---	---
9	Payload length	Integer	---	---
	If bit 0 of the extension code is set:			
10	Packet header LT address	Integer	---	---
11	Packet header type as numeric value	See definition	---	---
12	Packet header flow bit	Integer	---	---
13	Packet ARQN bit	Integer	---	---
14	Packet header SEQN bit	Integer	---	---
15	Packet header HEC	Hexadecimal	---	---
16	Payload header LLID	Integer	---	---
17	Payload header flow	Integer	---	---
18	CRC status	See definition	---	---

	If bit 1 of the extension code is set:			
19	Packet type (from analysis of captured data)	See definition		
20	Payload type (from analysis of captured data)	See definition		
21	Payload length (from analysis of captured data)	Integer		
	Note: The above three fields are intended for use with header exclusion mode. If header exclusion mode is **not** enabled these fields will contain duplicates of information obtained from the packet header.			
	If bit 2 of the extension code is set:			
22	BLE data rate	See definition		
23	BLR coding scheme	See definition		
	If bit 3 of the extension code is set:			
24	Coding indicator (2-bit field)	Binary		
	If decoded standard is 802.11b		---	---
6	Data rate	Real number	Mbps	0.1 Mbps
7	PPDU format	See definition	---	
8	PSDU length	Integer	---	---
9	PLCP CRC status	See definition	---	---
	If decoded standard is 802.11a or 802.11g:			
6	Type of packet	See definition	---	---
7	Data rate	Real number	Mbps	0.1 Mbps
8	Coding rate	See definition	---	---
9	Coding type	See definition	---	---
10	Modulation type	See definition	---	---
11	PSDU length	Integer	---	---
12	L-SIG parity check	See definition	---	---
13	Number of symbols	Integer	---	---
14	Number of tones	Integer	---	---
15	Guard interval	See definition	---	---
16	No. of signals	Integer	---	---
17	No. of spatial streams	Integer	---	---
18	No. of space time streams	Integer	---	---
	If decoded standard is 802.11p			
6	Type of packet (always LEGACY for p)	See definition		
7	Data rate	Real number	Mbps	0.1 Mbps
8	Coding rate	See definition	---	---
9	Coding type	See definition	---	---
10	Modulation type	See definition	---	---
11	PSDU length	Integer	---	---
12	L-SIG parity check	See definition	---	---

13	Number of symbols	Integer	---	---
14	Number of tones	Integer	---	---
15	Guard interval	See definition	---	---
16	No. of signals	Integer	---	---
17	No. of spatial streams	Integer	---	---
18	No. of space time streams	Integer	---	---
19	Bandwidth (5 MHz, 10 MHz or 20 MHz)	Integer	MHz	1 MHz

If decoded standard is 802.11n:

6	Type of packet	See definition	---	---
7	Data rate	Real number	Mbps	0.1 Mbps
8	Coding rate	See definition	---	---
9	Coding type	See definition	---	---
10	Modulation type	See definition	---	---
11	PSDU length	Integer	---	---
12	L-SIG parity check	See definition	---	---
13	Number of symbols	Integer	---	---
14	Number of tones	Integer	---	---
15	Guard interval	See definition	---	---
16	No. of signals	Integer	---	---
17	No. of spatial streams	Integer	---	---
18	No. of space time streams	Integer	---	---
19	MCS	Integer	---	---
20	PPDU type	See definition	---	---
21	PPDU format	See definition	---	---
22	HT-SIG CRC	See definition	---	---

If decoded standard is 802.11ac

6	Type of packet	See definition	---	---
7	Data rate	Real number	Mbps	0.1 Mbps
8	Coding rate	See definition	---	---
9	Coding type	See definition	---	---
10	Modulation type	See definition	---	---
11	PSDU length	Integer	---	---
12	L-SIG parity check	See definition	---	---
13	Number of symbols	Integer	---	---
14	Number of tones	Integer	---	---
15	Guard interval	See definition	---	---
16	No. of signals	Integer	---	---
17	No. of spatial streams	Integer	---	---
18	No. of space time streams	Integer	---	---
19	MCS	Integer	---	---
20	PPDU type	See definition	---	---
21	VHT-SIG CRC	See definition	---	---

Parameters

#	Description	Format	Units	Min	Max	Default
1	Segment index	Integer	---	---	---	---
2	Extension code	Integer	---	0	2^{32}-1	0

Decoded standard:	
WLA	WLAN 802.11a
WLB	WLAN 802.11b
WLG	WLAN 802.11g
WLP	WLAN 802.11p
WLN	WLAN 802.11n
WLAC	WLAN 802.11ac
BT	Bluetooth basic rate and EDR
BLE	Bluetooth low energy

CRC Status / parity checks:
PASS
FAIL

Bluetooth packet type:	
DH1, DH3, DH5	Basic Rate
2DH1, 2DH3, 2DH5	EDR 2 Mbps
3DH1, 3DH3, 3DH5,	EDR 3 Mbps
NULL, POLL,	Link management packets
BLE, 2LE, BLR	Bluetooth Low Energy
UNKNOWN	None of the above

Note: the above packet types can be measured. If the response is 'UNKNOWN', refer to the Bluetooth packet header type (as numeric value) field. This may indicate that the instrument has received a packet that is unsupported for measurement.

Bluetooth packet header type (as numeric value)	
0	NULL
1	POLL
2	FHS
3	DM1
4	DH1 or 2-DH1
5	HV1
6	HV2 or 2-EV3
7	HV3 or EV3 or 3-EV3
8	DV or 3-DH1

9	AUX1
10	DM3 or 2-DH3
11	DH3 or 3-DH3
12	EV4 or 2-EV5
13	EV5 or 3-EV5
14	DM5 or 2-DH5
15	DH5 or 3-DH5

Bluetooth payload type:

ZEROS	00000000
AAHEX	10101010
FZEROHEX	11110000
PRBS9	Pseudo random binary sequence (9^{th} order)
55HEX	01010101
ZEROFHEX	00001111
PRBS15	Pseudo random binary sequence (15^{th} order)
ONES	11111111
UNKNOWN	None of the above

Bluetooth low energy data rate:

125KBPS	125 kbps (BLR S=8)
500KBPS	500 kbps (BLR S=2)
1MBPS	1 Mbps (BLE)
2MBPS	2 Mbps (2LE)

BLR coding scheme:

S2	BLR S=2 coding
S8	BLR S=8 coding

WLAN packet type:

LEGACY	802.11g, a, p
HT	802.11n
VHT	802.11ac

Coding rate:

CR12	1/2
CR34	3/4
CR56	5/6

Coding type:

BCC	Binary convolutional coding
LDPC	Low density parity check

Modulation type:

BPSK	Binary phase shift keying
QPSK	Quadrature phase shift keying
QAM16	16-QAM (Quadrature amplitude modulation)
QAM64	64-QAM (Quadrature amplitude modulation)
QAM256	256-QAM (Quadrature amplitude modulation)

Guard interval:

LONG	Long guard interval (800 ns)
SHORT	Short guard interval (400 ns)

PPDU type (802.11n):

M20	Data is transmitted over a signal with 20 MHz bandwidth
M40	Data is transmitted over two adjacent signals with 20 MHz bandwidth
M40UPPER	Data is transmitted over a signal with 20 MHz bandwidth in the upper sideband of a 40 MHz channel
M40LOWER	Data is transmitted over a signal with 20 MHz bandwidth in the lower sideband of a 40 MHz channel.
M40DUP	Data is transmitted twice using two adjacent signals with 20 MHz bandwidth

PPDU type (802.11ac):

VHT20	Data is transmitted over a signal with 20 MHz bandwidth
VHT40	Data is transmitted over a signal with 40 MHz bandwidth
VHT80	Data is transmitted over a signal with 80 MHz bandwidth
VHT80_80	Data is transmitted over two signals with 80 MHz bandwidth
VHT160	Data is transmitted over a signal with 160 MHz bandwidth

PPDU format (802.11n):

MIXED	HT Mixed
GREENFIELD	HT Greenfield

PPDU Format (802.11b):

LONG

SHORT

Example of Use

```
:FETC:SRW:SEGM:IDEN? 1, 0
```

:FETCh:SRWireless:SUMMary:BLE:CDRift?

Function

Fetch *Bluetooth* LE measurements - carrier frequency offset and drift (TRM-LE/CA/BV-06-C, TRM-LE/CA/BV-12-C and TRM-LE/CA/BV-14-C)

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BLE:CDRift?
```

Response

#	Description	Format	Units	Res
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
4	Average frequency offset f_n (for segment)	Real number	Hz	1 Hz
5	Average drift $\lvert f_n - f_0 \rvert$ (for segment)	Real number	Hz	1 Hz
6	Average drift rate $\lvert f_n - f_{n-5} \rvert$ (for segment)	Real number	Hz	1 Hz
7	Worst case maximum frequency offset limit test passed	Boolean	---	---
8	Worst case maximum frequency drift limit test passed	Boolean	---	---
9	Worst case maximum drift rate test passed	Boolean	---	---
10	Worst case maximum frequency offset f_n	Real number	Hz	1 Hz
11	Worst case maximum frequency drift $\lvert f_n - f_0 \rvert$	Real number	Hz	1 Hz
12	Worst case maximum drift rate $\lvert f_n - f_{n-5} \rvert$	Real number	Hz	1 Hz
13	Packet number of worst case maximum frequency offset	Integer	---	---
14	Packet number of worst case maximum frequency drift	Integer	---	---
15	Packet number of worst case maximum drift rate	Integer	---	---

Notes on SIG Compliant and Speed Test modes (see :CONFigure:SRWireless:BT:MODE):

1. If the mode is SPEED, all response fields will be populated with results.
2. If the mode is STANDARD, some response fields will be populated with results depending on whether the measurement can be made with the specified packet payload according to the SIG Specification.
3. If a measurement cannot be made in STANDARD mode, the measurement field will contain the value NAN, 'not a number' (defined as 9.91E+37), any associated Boolean flags will be set to 0 (false), and any associated packet number (for example 'Packet number for Δf2 max worst case') will be set to zero.

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BLE:CDR? 1
```

:FETCh:SRWireless:SUMMary:BLE:CDRift? (Extended)

Function

Fetch *Bluetooth* LE measurements - carrier frequency offset and drift (TRM-LE/CA/BV-06-C, TRM-LE/CA/BV-12-C and TRM-LE/CA/BV-14-C). An optional extension code returns measurements in addition to those required by the standard.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BLE:CDRift?
```

Response

#	Description	Format	Units	Res
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code	Integer	---	---
4	Average frequency offset f_n (for segment)	Real number	Hz	1 Hz
5	Average drift $\lvert f_n - f_0 \rvert$ (for segment)	Real number	Hz	1 Hz
6	Average drift rate $\lvert f_n - f_{n-5} \rvert$ (for segment)	Real number	Hz	1 Hz
7	Worst case maximum frequency offset limit test passed	Boolean	---	---
8	Worst case maximum frequency drift limit test passed	Boolean	---	---
9	Worst case maximum drift rate test passed	Boolean	---	---
10	Worst case maximum frequency offset f_n	Real number	Hz	1 Hz
11	Worst case maximum frequency drift $\lvert f_n - f_0 \rvert$	Real number	Hz	1 Hz
12	Worst case maximum drift rate $\lvert f_n - f_{n-5} \rvert$	Real number	Hz	1 Hz
13	Packet number of worst case maximum frequency offset	Integer	---	---
14	Packet number of worst case maximum frequency drift	Integer	---	---
15	Packet number of worst case maximum drift rate	Integer	---	---
	If bit 0 of Extension Code is set:			
16	Average initial frequency offset f_0 (for segment)	Real number	Hz	1 Hz
17	Average initial frequency drift rate $\lvert f_1 - f_0 \rvert$ (for segment)	Real number	Hz	1 Hz
18	Worst case initial frequency offset f_0	Real number	Hz	1 Hz
19	Worst case initial frequency drift rate $\lvert f_1 - f_0 \rvert$	Real number	Hz	1 Hz
20	Worst case maximum positive frequency offset f_n	Real number	Hz	1 Hz
21	Worst case maximum negative frequency offset f_n	Real number	Hz	1 Hz
22	Worst case initial frequency offset limit test passed	Boolean	---	---
23	Worst case initial frequency drift rate limit test passed	Boolean	---	---

5

Remote Operation

24	Packet number of worst case initial frequency offset	Integer	---	---
25	Packet number of worst case initial frequency drift	Integer	---	---
26	Packet number of worst case maximum positive frequency offset	Integer	---	---
27	Packet number of worst case maximum negative frequency offset	Integer	---	---

Definitions from BLE Standard:

f_0 Carrier frequency offset measured over preamble.

f_1 Carrier frequency offset measured over first 10-bit block of payload.

f_n Carrier frequency offset measured over the n^{th} 10-bit block of payload.

$|f_n - f_{n-5}|$ Drift rate. This is the absolute value of the difference in carrier frequency offset between a 10-bit block in the payload and that measured five blocks (50 μs) earlier in the payload. In other words, this is the drift rate over a 50 μs time period within the payload. The instrument measures the worst case drift rate over the payload.

$|f_1 - f_0|$ Initial carrier drift. The absolute value of the difference in carrier frequency offset between the first 10-bit block in the payload and that of the preamble.

$|f_0 - f_n|$ Carrier drift. The absolute value of the difference in carrier frequency offset between any 10-bit block in the payload and that of the preamble. The instrument measures the average and worst case drift across the packet payload.

Note on limit checking extended measurements:

The initial frequency offset limit is 150 kHz by default and may be changed using `:CONF:SRW:BLE:CFOF:LIM`.
The initial frequency drift rate limit is 50 kHz by default and may be changed using `:CONF:SRW:BLE:DRAT:LIM`.

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:BLE:CDR? 1,1
```

:FETCh:SRWireless:SUMMary:BLE:DEViation:ASCii?

Function

Fetch *Bluetooth* measurements - deviation vs symbol (BLE) – ASCII format

Command

N/A

Query

`:FETCh:SRWireless:SUMMary:BLE:DEViation:ASCii?`

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	
2	Packet index	Integer	---	
3	Start time	Real number	s	1 μs
4	Stop time	Real number	s	1 μs
5	Number of points (n)	Integer	---	
6	First deviation point (averaged over segment)	Real number	Hz	1 Hz
---	...			
	Last deviation point (averaged over segment)	Real number	Hz	1 Hz
	Index fields valid	Boolean	---	---
	Start offset	Integer	---	---
	Stop offset	Integer	---	---
	Preamble index (p0)	Integer	---	---
	Sync word index	Integer	---	---
	PDU header index	Integer	---	---
	PDU length index	Integer	---	---
	PDU payload index	Integer	---	---
	CRC start index	Integer	---	---
	CRC stop index	Integer	---	---

Notes:

1. The index fields contain offsets from the start of the deviation v symbol array that identify the positions of specific fields. A typical use is for plotting deviation v symbol with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BLE:DEV:ASC? 1
```

:FETCh:SRWireless:SUMMary:BLE:DEViation[:BINary]?

Function

Fetch *Bluetooth* measurements - deviation vs symbol (BLE) – binary format

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BLE:DEViation[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Segment index	INT32	---
3	Packet index (always zero)	INT32	---
4	Start time (relative to p0)	REAL32	s
5	Stop time (relative to p0)	REAL32	s
6	Number of points (n)	INT32	---
7	Position of p0 relative to start of power data	INT32	---
8	Deviation in Hz	REAL32	Hz
…	Index fields valid if non zero.		
	Start offset (for use when zooming)	REAL32	Hz
	Stop offset (for use when zooming)	INT32	---
	Index position of preamble	INT32	---
	Index position of sync word	INT32	---
	Index position of PDU header	INT32	---
	Index position of PDU length field	INT32	---
	Index position of PDU payload	INT32	---
	Index position of CRC start	INT32	---
	Index position of CRC stop	INT32	---

Notes:

1. The index fields contain offsets from the start of the deviation v symbol array that identify the positions of specific fields. A typical use is for plotting deviation v symbol with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.
4. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
5. Binary data will always be a multiple of 4 bytes.
6. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BLE:DEV? 1
```

:FETCh:SRWireless:SUMMary:BLE:IBEMissions?

Function

Fetch Bluetooth LE in-band emissions measurements for 1 Mbps and 2 Mbps (2LE) (TRM-LE/CA/BV-03-C and TRM-LE/CA/BV-08-C). BLR signals are also supported although test limits are not defined.

To return a full set of results for 2LE measurements an extension code is required.

Extension code bit	Additional fields in response
0	Additional results at ±6, 7 and 8 MHz from the transmit frequency as required when measuring 2LE signals.

Notes:
The following table defines which of the response fields contain valid results, depending on whether the signal being measured is BLE (1 Mbps), 2LE (2 Mbps) or BLR. The value NaN, 'not a number' is defined as 9.91E+37.

<table>
<tr><th></th><th colspan="3">Bluetooth Low Energy Packet Type</th></tr>
<tr><th>Offset from TX frequency</th><th>(Original) BLE
1 Mbps</th><th>2LE
2 Mbps</th><th>BLR
Long Range</th></tr>
<tr><td>±8 MHz</td><td rowspan="3">NaN returned for power measurements and "PASS" returned for limit tests.
Worst case packet number fields return 0.</td><td rowspan="3">Power results are returned and checked against limits.</td><td rowspan="3">NaN returned for power measurements and "PASS" results for limit tests.
Worst case packet number fields return 0.</td></tr>
<tr><td>±7 MHz</td></tr>
<tr><td>±6 MHz</td></tr>
<tr><td>±5 MHz</td><td rowspan="4">Power results are returned and checked against limits.</td><td rowspan="2">Power results are returned and checked against limits</td><td rowspan="6">Power results are returned but limit results are always "PASS" because the Bluetooth standard does not specify limits for BLR signals.</td></tr>
<tr><td>±4 MHz</td></tr>
<tr><td>±3 MHz</td><td rowspan="4">Power results are returned but limit results are always "PASS" because the Bluetooth standard does not specify limits at these frequency offsets.</td></tr>
<tr><td>±2 MHz</td></tr>
<tr><td>±1 MHz</td><td rowspan="2">Power results are returned but limit results are always "PASS" because the Bluetooth standard does not specify limits at these frequency offsets.</td></tr>
<tr><td>0 Hz</td></tr>
</table>

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BLE:IBEMissions?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code	Integer	---	---
	If bit 0 of extension code is set:			
4	TX frequency – 8 MHz power (worst packet)	Real number	dBm	0.01 dB
5	TX frequency – 7 MHz power (worst packet)	Real number	dBm	0.01 dB
6	TX frequency – 6 MHz power (worst packet)	Real number	dBm	0.01 dB
7	TX frequency – 5 MHz power (worst packet)	Real number	dBm	0.01 dB
8	TX frequency – 4 MHz power (worst packet)	Real number	dBm	0.01 dB
9	TX frequency – 3 MHz power (worst packet)	Real number	dBm	0.01 dB
10	TX frequency – 2 MHz power (worst packet)	Real number	dBm	0.01 dB
11	TX frequency – 1 MHz power (worst packet)	Real number	dBm	0.01 dB
12	TX frequency power (from packet with highest power)	Real number	dBm	0.01 dB
13	TX frequency + 1 MHz power (worst packet)	Real number	dBm	0.01 dB
14	TX frequency + 2 MHz power (worst packet)	Real number	dBm	0.01 dB
15	TX frequency + 3 MHz power (worst packet)	Real number	dBm	0.01 dB
16	TX frequency + 4 MHz power (worst packet)	Real number	dBm	0.01 dB
17	TX frequency + 5 MHz power (worst packet)	Real number	dBm	0.01 dB
	If bit 0 of extension code is set:			
18	TX frequency + 6 MHz power (worst packet)	Real number	dBm	0.01 dB
19	TX frequency + 7 MHz power (worst packet)	Real number	dBm	0.01 dB
20	TX frequency + 8 MHz power (worst packet)	Real number	dBm	0.01 dB
21	Overall test passed	Boolean	---	---
	If bit 0 of extension code is set:			
22	TX frequency – 8 MHz test passed	Boolean	---	---
23	TX frequency – 7 MHz test passed	Boolean	---	---
24	TX frequency – 6 MHz test passed	Boolean	---	---
25	TX frequency – 5 MHz test passed	Boolean	---	---
26	TX frequency – 4 MHz test passed	Boolean	---	---
27	TX frequency – 3 MHz test passed	Boolean	---	---
28	TX frequency – 2 MHz test passed	Boolean	---	---
29	TX frequency – 1 MHz test passed	Boolean	---	---
30	TX frequency test passed	Boolean	---	---
31	TX frequency + 1 MHz test passed	Boolean	---	---
32	TX frequency + 2 MHz test passed	Boolean	---	---
33	TX frequency + 3 MHz test passed	Boolean	---	---
34	TX frequency + 4 MHz test passed	Boolean	---	---
35	TX frequency + 5 MHz test passed	Boolean	---	---

	If bit 0 of extension code is set:			
36	TX frequency + 6 MHz test passed	Boolean	---	---
37	TX frequency + 7 MHz test passed	Boolean	---	---
38	TX frequency + 8 MHz test passed	Boolean	---	---
	If bit 0 of extension code is set:			
39	TX frequency – 8 MHz worst case packet number	Integer	---	---
40	TX frequency – 7 MHz worst case packet number	Integer	---	---
41	TX frequency – 6 MHz worst case packet number	Integer	---	---
42	TX frequency – 5 MHz worst case packet number	Integer	---	---
43	TX frequency – 4 MHz worst case packet number	Integer	---	---
44	TX frequency – 3 MHz worst case packet number	Integer	---	---
45	TX frequency – 2 MHz worst case packet number	Integer	---	---
46	TX frequency – 1 MHz worst case packet number	Integer	---	---
47	Packet number of packet with highest channel power	Integer	---	---
48	TX frequency + 1 MHz worst case packet number	Integer	---	---
49	TX frequency + 2 MHz worst case packet number	Integer	---	---
50	TX frequency + 3 MHz worst case packet number	Integer	---	---
51	TX frequency + 4 MHz worst case packet number	Integer	---	---
52	TX frequency + 5 MHz worst case packet number	Integer	---	---
	If bit 0 of extension code is set:			
53	TX frequency + 6 MHz worst case packet number	Integer	---	---
54	TX frequency + 7 MHz worst case packet number	Integer	---	---
55	TX frequency + 8 MHz worst case packet number	Integer	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:BLE:IBEM? 1,1
```

Function

Fetch Bluetooth LE in-band emissions measurements for 1 Mbps and 2 Mbps (2LE) (TRM-LE/CA/BV-03-C and TRM-LE/CA/BV-07-C). BLR signals are also supported although test limits are not defined.

To return a full set of results for 2LE measurements an extension code is required.

Extension code bit	Additional fields in response
0	Additional results at ±6, 7 and 8 MHz from the transmit frequency as required when measuring 2LE signals.

Notes:

The following table defines which of the response fields contain valid results, depending on whether the signal being measured is BLE (1 Mbps), 2LE (2 Mbps) or BLR. The value NaN, ‘not a number’ is defined as 9.91E+37.

<table>
<tr><th></th><th colspan="3">Bluetooth Low Energy Packet Type</th></tr>
<tr><th>Offset from TX frequency</th><th>(Original) BLE
1 Mbps</th><th>2LE
2 Mbps</th><th>BLR
Long Range</th></tr>
<tr><td>±8 MHz</td><td rowspan="3">NaN returned for power measurements and “PASS” returned for limit tests.
Worst case packet number fields return 0.</td><td rowspan="3">Power results are returned and checked against limits.</td><td rowspan="3">NaN returned for power measurements and “PASS” results for limit tests.
Worst case packet number fields return 0.</td></tr>
<tr><td>±7 MHz</td></tr>
<tr><td>±6 MHz</td></tr>
<tr><td>±5 MHz</td><td rowspan="4">Power results are returned and checked against limits.</td><td rowspan="2">Power results are returned and checked against limits</td><td rowspan="7">Power results are returned but limit results are always “PASS” because the Bluetooth standard does not specify limits for BLR signals.</td></tr>
<tr><td>±4 MHz</td></tr>
<tr><td>±3 MHz</td><td rowspan="4">Power results are returned but limit results are always “PASS” because the Bluetooth standard does not specify limits at these frequency offsets.</td></tr>
<tr><td>±2 MHz</td></tr>
<tr><td>±1 MHz</td><td rowspan="2">Power results are returned but limit results are always “PASS” because the Bluetooth standard does not specify limits at these frequency offsets.</td></tr>
<tr><td>0 Hz</td></tr>
</table>

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BLE:IBEMissions?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code	Integer	---	---
	If bit 0 of extension code is set:			
4	TX frequency – 8 MHz power (worst packet)	Real number	dBm	0.01 dB
5	TX frequency – 7 MHz power (worst packet)	Real number	dBm	0.01 dB
6	TX frequency – 6 MHz power (worst packet)	Real number	dBm	0.01 dB
7	TX frequency – 5 MHz power (worst packet)	Real number	dBm	0.01 dB
8	TX frequency – 4 MHz power (worst packet)	Real number	dBm	0.01 dB
9	TX frequency – 3 MHz power (worst packet)	Real number	dBm	0.01 dB
10	TX frequency – 2 MHz power (worst packet)	Real number	dBm	0.01 dB
11	TX frequency – 1 MHz power (worst packet)	Real number	dBm	0.01 dB
12	TX frequency power (from packet with highest power)	Real number	dBm	0.01 dB
13	TX frequency + 1 MHz power (worst packet)	Real number	dBm	0.01 dB
14	TX frequency + 2 MHz power (worst packet)	Real number	dBm	0.01 dB
15	TX frequency + 3 MHz power (worst packet)	Real number	dBm	0.01 dB
16	TX frequency + 4 MHz power (worst packet)	Real number	dBm	0.01 dB
17	TX frequency + 5 MHz power (worst packet)	Real number	dBm	0.01 dB
	If bit 0 of extension code is set:			
18	TX frequency + 6 MHz power (worst packet)	Real number	dBm	0.01 dB
19	TX frequency + 7 MHz power (worst packet)	Real number	dBm	0.01 dB
20	TX frequency + 8 MHz power (worst packet)	Real number	dBm	0.01 dB
21	Overall test passed	Boolean	---	---
	If bit 0 of extension code is set:			
22	TX frequency – 8 MHz test passed	Boolean	---	---
23	TX frequency – 7 MHz test passed	Boolean	---	---
24	TX frequency – 6 MHz test passed	Boolean	---	---
25	TX frequency – 5 MHz test passed	Boolean	---	---
26	TX frequency – 4 MHz test passed	Boolean	---	---
27	TX frequency – 3 MHz test passed	Boolean	---	---
28	TX frequency – 2 MHz test passed	Boolean	---	---
29	TX frequency – 1 MHz test passed	Boolean	---	---
30	TX frequency test passed	Boolean	---	---
31	TX frequency + 1 MHz test passed	Boolean	---	---
32	TX frequency + 2 MHz test passed	Boolean	---	---
33	TX frequency + 3 MHz test passed	Boolean	---	---
34	TX frequency + 4 MHz test passed	Boolean	---	---
35	TX frequency + 5 MHz test passed	Boolean	---	---
	If bit 0 of extension code is set:			
36	TX frequency + 6 MHz test passed	Boolean	---	---
37	TX frequency + 7 MHz test passed	Boolean	---	---
38	TX frequency + 8 MHz test passed	Boolean	---	---

	If bit 0 of extension code is set:			
39	TX frequency – 8 MHz worst case packet number	Integer	---	---
40	TX frequency – 7 MHz worst case packet number	Integer	---	---
41	TX frequency – 6 MHz worst case packet number	Integer	---	---
42	TX frequency – 5 MHz worst case packet number	Integer	---	---
43	TX frequency – 4 MHz worst case packet number	Integer	---	---
44	TX frequency – 3 MHz worst case packet number	Integer	---	---
45	TX frequency – 2 MHz worst case packet number	Integer	---	---
46	TX frequency – 1 MHz worst case packet number	Integer	---	---
47	Packet number of packet with highest channel power	Integer	---	---
48	TX frequency + 1 MHz worst case packet number	Integer	---	---
49	TX frequency + 2 MHz worst case packet number	Integer	---	---
50	TX frequency + 3 MHz worst case packet number	Integer	---	---
51	TX frequency + 4 MHz worst case packet number	Integer	---	---
52	TX frequency + 5 MHz worst case packet number	Integer	---	---
	If bit 0 of extension code is set:			
53	TX frequency + 6 MHz worst case packet number	Integer	---	---
54	TX frequency + 7 MHz worst case packet number	Integer	---	---
55	TX frequency + 8 MHz worst case packet number	Integer	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:BLE:IBEM? 1,1
```

:FETCh:SRWireless:SUMMary:BLE:MODulation?

Function

Fetch *Bluetooth* LE measurements - modulation characteristics (TRM-LE/CA/BV-05-C, TRM-LE/CA/BV-09-C, TRM-LE/CA/BV-10-C, TRM-LE/CA/BV-11-C, and TRM-LE/CA/BV-13-C)

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BLE:MODulation?
```

Response

The following responses apply for measurements on BLE (1 Mbps), 2LE (2 Mbps) and BLR S=2 packets:

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Average of Δf2 avg	Real number	Hz	100 Hz
4	Δf2 max worst case	Real number	Hz	100 Hz
5	Average of Δf1 avg	Real number	Hz	100 Hz
6	Δf1 max worst case	Real number	Hz	100 Hz
7	(Δf2 avg / Δf1 avg) worst case	Real number	---	0.01
8	%Δf2 max > 115 kHz worst case	Real number	%	0.1 %
9	Δf1 avg lower limit test passed	Boolean	---	---
10	Δf1 avg upper limit test passed	Boolean	---	---
11	Δf2 avg / Δf1 avg limit test passed	Boolean	---	---
12	%Δf2 max > 115 kHz limit test passed	Boolean	---	---
13	Packet number for Δf2 max worst case	Integer	---	---
14	Packet number for Δf1 max worst case	Integer	---	---
15	Packet number for (Δf2 avg / Δf1 avg) worst case	Integer	---	---
16	Packet number for %Δf2 max > 115 kHz worst case	Integer	---	---

The following responses apply for measurements on BLR S=8 packets:

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	NaN (Δf2 measurements are not possible on BLR S=8 packets)	Real number		
4	Δf1 max – lowest value - measured across all packets.	Real number	Hz	100 Hz
5	Average of Δf1 avg	Real number	Hz	100 Hz
6	Δf1 max – peak value - worst case	Real number	Hz	100 Hz

7	NaN (Δf2 avg / Δf1 avg is not applicable for BLR S=8)	Real number	---	0.01
8	%Δf1 max > 185 kHz worst case	Real number	%	0.1 %
9	Δf1 avg lower limit test passed	Boolean	---	---
10	Δf1 avg upper limit test passed	Boolean	---	---
11	0 (false) (Δf2 avg / Δf1 avg test is not performed for BLR S-8)	Boolean	---	---
12	%Δf2 max > 185 kHz limit test passed	Boolean	---	---
13	Packet number for Δf1 max – lowest value worst case	Integer	---	---
14	Packet number for Δf1 max – peak value worst case	Integer	---	---
15	0 (Δf2 avg / Δf1 avg test is not performed for BLR S=8)	Integer	---	---
16	Packet number for %Δf2 max > 185 kHz worst case	Integer	---	---

Notes on SIG Compliant and Speed Test modes (see :CONFigure:SRWireless:BLE:MODE):

1. If the mode is SPEED, all response fields will be populated with results (but see the notes for BLR S=8 packets below).
2. If the mode is STANDARD, some response fields will be populated with results depending on whether the measurement can be made with the specified packet.
3. If a measurement cannot be made in STANDARD mode, the measurement field will contain the value NAN, 'not a number' (defined as 9.91E+37), any associated Boolean flags will be set to 0 (false), and any associated packet number will be set to zero.

Notes on measurements on BLR S=2 and S=8 packets:

1. TRM-LE/CA/BV-12-C specifies measurements on BLR S=8 packets having a 11111111 payload which, after encoding, produces a repeating 00111100 pattern suitable for measurement. In STANDARD mode BLR S=2 packets and BLR S=8 packets containing a payload other than 11111111 will not return results.
2. BLR S=2 and S=8 packets will return measurements in SPEED mode.
3. BLR S=8 packets do not include 0101 patterns, so “Δf2” measurements are not returned.
4. If a measurement cannot be made the measurement field will contain the value NaN, 'not a number' (defined as 9.91E+37), any associated Boolean flags will be set to 0 (false), and any associated packet number (for example 'Packet number for Δf2 max worst case') will be set to zero.

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BLE:MOD? 1
```

:FETCh:SRWireless:SUMMary:BLE:POWer?

Function

Fetch *Bluetooth* LE measurements - output power (TRM-LE/CA/BV-01-C).

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BLE:POWer?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Average of packet average powers for segment	Real number	dBm	0.01 dB
4	Maximum crest factor for segment	Real number	dB	0.01 dB
5	Average power lower limit test passed	Boolean	---	---
6	Average power upper limit test passed	Boolean	---	---
7	Crest factor limit test passed	Boolean	---	---
8	Minimum crest factor for segment	Real number	dB	0.01 dB
9	Packet number containing maximum crest factor	Integer	---	---
10	Packet number containing minimum crest factor	Integer	---	---

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BLE:POW? 1
```

:FETCh:SRWireless:SUMMary:BLE:PPRofile:AVERage:ASCii?

Function

Fetch *Bluetooth* measurements - power burst profile (BLE) – average data – ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BLE:PPRofile:AVERage:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	
2	Packet index (always zero)	Integer	---	
3	Start time (relative to p0)	Real number	s	1 µs
4	Stop time (relative to p0)	Real number	s	1 µs
5	Number of points (n)	Integer	---	
6	First average power profile point	Real number	dBm	0.01 dB
---	…			
	Last average power profile point	Real number	dBm	0.01 dB
	Index fields valid	Boolean	---	---
	Start offset	Integer	---	---
	Stop offset	Integer	---	---
	Preamble index (p0)	Integer	---	---
	Sync word index	Integer	---	---
	PDU header index	Integer	---	---
	PDU length index	Integer	---	---
	PDU payload index	Integer	---	---
	CRC start index	Integer	---	---
	CRC stop index	Integer	---	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile. A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BLE:PPR:AVER:ASC? 1
```

:FETCh:SRWireless:SUMMary:BLE:PPRofile:AVERage[:BINary]?

Function

Fetch *Bluetooth* measurements - power burst profile (BLE) – average data – binary format.

Command

N/A

Query

`:FETCh:SRWireless:SUMMary:BLE:PPRofile:AVERage[:BINary]?`

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Segment index	INT32	---
3	Packet index (always zero)	INT32	---
4	Start time (relative to p0)	REAL32	s
5	Stop time (relative to p0)	REAL32	s
6	Number of points (n)	INT32	---
7	Power in dBm	REAL32	dBm
…	…		
	Index fields (valid if non-zero)	INT32	---
	Start offset (for use when zooming)	INT32	---
	Stop offset (for use when zooming)	INT32	---
	Index position of preamble	INT32	---
	Index position of sync word	INT32	---
	Index position of PDU header	INT32	---
	Index position of PDU length field	INT32	---
	Index position of PDU payload	INT32	---
	Index position of CRC start	INT32	---
	Index position of CRC stop	INT32	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile. A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.
4. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
5. Binary data will always be a multiple of 4 bytes.

6. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BLE:PPR:AVER? 1
```

:FETCh:SRWireless:SUMMary:BLE:PPRofile:PEAK:ASCii?

Function

Fetch *Bluetooth* measurements - power burst profile (BLE) – peak data – ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BLE:PPRofile:PEAk:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	
2	Packet index	Integer	---	
3	Start time (relative to p0)	Real number	s	1 µs
4	Stop time (relative to p0)	Real number	s	1 µs
5	Number of points (n)	Integer	---	
6	First peak power profile point	Real number	dBm	0.01 dB
---	...			
	Last peak power profile point	Real number	dBm	0.01 dB
	Index fields valid	Boolean	---	---
	Start offset	Integer	---	---
	Stop offset	Integer	---	---
	Preamble index (p0)	Integer	---	---
	Sync word index	Integer	---	---
	PDU header index	Integer		
	PDU length index	Integer		
	PDU payload index	Integer		
	CRC start index	Integer	---	---
	CRC stop index	Integer	---	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile. A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BLE:PPR:PEAK:ASC? 1
```

:FETCh:SRWireless:SUMMary:BLE:PPRofile:PEAK[:BINary]?

Function

Fetch *Bluetooth* measurements - power burst profile (BLE) – peak data – binary format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BLE:PPRofile:PEAK[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Segment index	INT32	---
3	Packet index (always zero)	INT32	---
4	Start time (relative to p0)	REAL32	s
5	Stop time (relative to p0)	REAL32	s
6	Number of points (n)	INT32	---
7	Power in dBm	REAL32	dBm
…	…		
	Index fields valid if non zero.	INT32	---
	Start offset (for use when zooming)	INT32	---
	Stop offset (for use when zooming)	INT32	---
	Index position of preamble	INT32	---
	Index position of sync word	INT32	---
	Index position of PDU header	INT32	---
	Index position of PDU length field	INT32	---
	Index position of PDU payload	INT32	---
	Index position of CRC start	INT32	---
	Index position of CRC stop	INT32	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile. A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.
4. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
5. Binary data will always be a multiple of 4 bytes.

6. Refer to sections 5.3.3 and 5.3.4 *Binary_Header_Information* and *Binary_Data_Fields*.

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BLE:PPR:PEAK? 1
```

:FETCh:SRWireless:SUMMary:BT:ACPower?

Function

Fetch the *Bluetooth* adjacent channel power measurements (TRM/CA/06/C).

Command

N/A

Query

`:FETCh:SRWireless:SUMMary:BT:ACPower?`

Response

#	Description	Format	Units
1	Segment index	Integer	---
2	Packet index (always zero)	Integer	---
3	Channel–5 power	Real Number	dBm
4	Channel–4 power	Real Number	dBm
5	Channel–3 power	Real Number	dBm
6	Channel–2 power	Real Number	dBm
7	Channel–1 power	Real Number	dBm
8	Channel power	Real Number	dBm
9	Channel+1 power	Real Number	dBm
10	Channel+2 power	Real Number	dBm
11	Channel+3 power	Real Number	dBm
12	Channel+4 power	Real Number	dBm
13	Channel+5 power	Real Number	dBm
14	Overall test passed	Boolean	---
15	Channel–5 test passed	Boolean	---
16	Channel–4 test passed	Boolean	---
17	Channel–3 test passed	Boolean	---
18	Channel–2 test passed	Boolean	---
19	Channel–1 test passed (always pass)	Boolean	---
20	Channel test passed (always pass)	Boolean	---
21	Channel+1 test passed (always pass)	Boolean	---
22	Channel+2 test passed	Boolean	---
23	Channel+3 test passed	Boolean	---
24	Channel+4 test passed	Boolean	---
25	Channel+5 test passed	Boolean	---
26	Channel–5 worst case packet number	Integer	---
27	Channel–4 worst case packet number	Integer	---
28	Channel–3 worst case packet number	Integer	---
29	Channel–2 worst case packet number	Integer	---
30	Channel–1 worst case packet number	Integer	---

31	Channel worst case packet number	Integer	---
32	Channel+1 worst case packet number	Integer	---
33	Channel+2 worst case packet number	Integer	---
34	Channel+3 worst case packet number	Integer	---
35	Channel+4 worst case packet number	Integer	---
36	Channel+5 worst case packet number	Integer	---

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:ACP? 1
```

:FETCh:SRWireless:SUMMary:BT:BANDwidth?

Function

Fetch *Bluetooth* 20 dB bandwidth measurement results (TRM/CA/05/C).

Currently no extension codes have been defined for this command. Use zero.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:BANDwidth?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	20 dB bandwidth (averaged over segment)	Integer	Hz	1 Hz
5	Bandwidth limit test passed	Boolean	---	---
6	Spectrum bandwidth (in practice always 3 MHz).	Integer	Hz	1 Hz
7	Number of points in spectrum	Integer	---	---
	For each point in spectrum:	Integer	---	---
	Power (averaged over segment)	Real number	dBm	0.01 dB
	Packet number of worst case packet	Integer	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:BT:BAND? 1,0
>1,0,0,914062,1,3000000,384,-69.49, … -55.68,1    // Segment 1
                                                  // Packet 0 (summary)
                                                  // Extension code 0
                                                  // 20 dB BW 914.062 kHz
                                                  // Limit test passed.
                                                  // Spectrum BW is 3 MHz
                                                  // 384 points in spectrum
                                                  // Power values…
                                                  // Packet 1 is worst case
```

5

Remote Operation

:FETCh:SRWireless:SUMMary:BT:CDRift?

Function

Fetch *Bluetooth* measurements - carrier drift (TRM/CA/09/C).

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:CDRift?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Average carrier frequency drift	Real number	Hz	Hz
4	Average carrier frequency drift rate	Real number	Hz	Hz
5	Carrier frequency drift limit test passed	Boolean	---	---
6	Carrier frequency drift rate test passed	Boolean	---	---
7	Worst case carrier frequency drift	Real number	Hz	Hz
8	Worst case carrier frequency drift rate	Real number	Hz	Hz
9	Packet number of worst case frequency drift.	Integer	---	---
10	Packet number of worst case frequency drift rate.	Integer	---	---

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:CDR? 1
```

:FETCh:SRWireless:SUMMary:BT:DEViation:ASCii?

Function

Fetch *Bluetooth* measurements - deviation vs symbol (BR) – ASCII format

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:DEViation:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	
2	Packet index (always zero)	Integer	---	
3	Start time (relative to p0)	Real number	s	1 μs
4	Stop time (relative to p0)	Real number	s	1 μs
5	Number of points (n)	Integer	---	
6	First deviation point (averaged over segment)	Real number	Hz	1 Hz
---	…			
	Last deviation point (averaged over segment)	Real number	Hz	1 Hz
	Index fields valid	Boolean	---	---
	Start offset (always zero)	Integer	---	---
	Stop offset (always zero)	Integer	---	---
	Preamble index (p0)	Integer	---	---
	Sync word index	Integer	---	---
	Trailer index	Integer	---	---
	Packet header index	Integer	---	---
	Guard time index	Integer	---	---
	EDR sync index	Integer	---	---
	Payload header index	Integer	---	---
	Payload index	Integer	---	---
	CRC start index	Integer	---	---
	CRC stop index	Integer	---	---

Notes:

1. The index fields contain offsets from the start of the deviation v symbol array that identify the positions of specific fields. A typical use is for plotting deviation v symbol with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:DEV:ASC? 1
```

:FETCh:SRWireless:SUMMary:BT:DEViation[:BINary]?

Function

Fetch *Bluetooth* measurements - deviation vs symbol (BR) – binary format

Command

N/A

Query

`:FETCh:SRWireless:SUMMary:BT:DEViation[:BINary]?`

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Segment index	INT32	---
3	Packet index (always zero)	INT32	---
4	Start time (relative to p0)	REAL32	s
5	Stop time (relative to p0)	REAL32	s
6	Number of points (n)	INT32	---
7	Position of p0 relative to start of power data	INT32	---
8	First deviation point (averaged over segment)	REAL32	Hz
…	…		
	Last deviation point (averaged over segment)	REAL32	Hz
	Index fields (valid if non-zero)	INT32	---
	Start offset (always zero)	INT32	---
	Stop offset (always zero)	INT32	---
	Preamble index (p0)	INT32	---
	Sync word index	INT32	---
	Trailer index	INT32	---
	Packet header index	INT32	---
	Guard time index	INT32	---
	EDR sync index	INT32	---
	Payload header index	INT32	---
	Payload index	INT32	---
	CRC start index	INT32	---
	CRC stop index	INT32	---

Notes:

1. The index fields contain offsets from the start of the deviation v symbol array that identify the positions of specific fields. A typical use is for plotting deviation v symbol with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.

5

Remote Operation

3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.
4. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
5. Binary data will always be a multiple of 4 bytes.
6. Refer to sections 5.3.3 and 5.3.4 *Binary_Header_Information* and *Binary_Data_Fields*.

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:DEV? 1
```

:FETCh:SRWireless:SUMMary:BT:DEVM:ASCii?

Function

Fetch *Bluetooth* measurements - DEVM v symbol (EDR) – ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:DEVM:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Number of points	Integer	---	---
4	First DEVM value	Real number	---	0.001
…				
	Last DEVM value	Real number	---	0.001

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:DEVM:ASC? 1
```

:FETCh:SRWireless:SUMMary:BT:DEVM[:BINary]?

Function

Fetch *Bluetooth* measurements - DEVM v symbol (EDR) – binary format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:DEVM[:BINary]?
```

Response

#	Description	Format	Units	Resolution
1	#nmmm…mm	Binary header	---	---
2	Segment index	INT32	---	---
3	Packet index (always zero)	INT32	---	---
4	Number of points	INT32	---	---
5	First DEVM value	REAL32	---	0.01
…	…	…	…	…
	Last DEVM value	REAL32	---	0.01

Parameters

#	Description	Format
1	Segment index	Number

Note:

1. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Example of Use

```
:FETC:SRW:SUMM:BT:DEVM? 1
```

:FETCh:SRWireless:SUMMary:BT:ECModulation?

Function

Fetch *Bluetooth* measurements - EDR carrier frequency stability and modulation accuracy (TRM/CA/11/C).

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:ECModulation
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer		---
3	Initial frequency error (ω_i) averaged across all packets	Real number	Hz	1 Hz
4	Frequency error (ω_0) averaged across all packets	Real number	Hz	1 Hz
5	Block frequency error ($\omega_i + \omega_0$) averaged for all packets	Real number	Hz	1 Hz
6	Payload modulation scheme	<CRD>	---	---
7	RMS DEVM averaged across all packets	Real number		0.001
8	Peak DEVM averaged across all packets	Real number		0.001
9	% symbols meeting DEVM limit averaged over all packets	Real number		0.01
10	Guard interval	Real number	s	50 ns
11	Carrier frequency stability limit test passed - all packets	Boolean	---	---
12	RMS DEVM limit test passed - all packets	Boolean	---	---
13	Peak DEVM limit test passed - all packets	Boolean	---	---
14	Percentage of symbols limit test passed	Boolean	---	---
15	Guard interval limit test passed - all packets	Boolean	---	---
16	Worst case ω_i	Real number	Hz	1 Hz
17	Worst case ω_0	Real number	Hz	1 Hz
18	Worst case block frequency error ($\omega_i + \omega_0$)	Real number	Hz	1 Hz
19	Worst case RMS DEVM	Real number		0.001
20	Worst case peak DEVM	Real number		0.001
21	% symbols meeting DEVM limit worst case.	Real number		0.01
22	Worst case guard interval	Real number	s	50 ns
23	Packet number for worst case ω_i	Integer	---	---
24	Packet number for worst case ω_0	Integer	---	---
25	Packet number for worst case ($\omega_i + \omega_0$)	Integer	---	---
26	Packet number for worst case RMS DEVM	Integer	---	---
27	Packet number for worst case peak DEVM	Integer		---

28	Packet number for worst case % symbols meeting limit	Integer	---
29	Packet number for worst case guard interval	Integer	---

Notes on SIG Compliant and Speed Test modes (see :CONFigure:SRWireless:BT:MODE):

1. If the mode is SPEED, all response fields will be populated with results.
2. If the mode is STANDARD, some response fields will be populated with results depending on whether the measurement can be made with the specified packet payload according to the SIG Specification. See Chapter 4 for a detailed description of *Bluetooth* measurements in SIG Standard mode.
3. If a measurement cannot be made in STANDARD mode, the measurement field will contain the value NAN, 'not a number' (defined as 9.91E+37), any associated Boolean flags will be set to 0 (false), and any associated packet number (for example 'Packet number for Δf2 max worst case') will be set to zero.

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:ECM? 1
```

:FETCh:SRWireless:SUMMary:BT:ECModulation? (Extended)

Function

Fetch *Bluetooth* measurements - EDR carrier frequency stability and modulation accuracy (TRM/CA/11/C).

The optional extension code adds additional results to the response as described in the following table:

Extension code bit	Additional fields in response
0	Individual limit test results for: • Initial frequency error (ω_i) • Frequency error (ω_0) • Block frequency error ($\omega_i + \omega_0$)

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:ECModulation?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code	Integer	---	---
4	Initial frequency error (ω_i) averaged across all packets	Real number	Hz	1 Hz
5	Frequency error (ω_0) averaged across all packets	Real number	Hz	1 Hz
6	Block frequency error ($\omega_i + \omega_0$) averaged for all packets	Real number	Hz	1 Hz
7	Payload modulation scheme	<CRD>	---	---
8	RMS DEVM averaged across all packets	Real number	%	0.001 %
9	Peak DEVM averaged across all packets	Real number	%	0.001 %
10	% symbols meeting DEVM limit averaged over all packets	Real number	%	0.01 %
11	Guard interval	Real number	s	50 ns
	If bit 0 of extension code set:			
12	Initial frequency error (ω_i) limit test passed	Boolean	---	---
13	Frequency error (ω_0) limit test passed	Boolean	---	---
14	Block frequency error ($\omega_i + \omega_0$) limit test passed	Boolean	---	---
15	Carrier frequency stability limit test passed - all packets. ***Note:*** *This is the logical AND of the limit test results for (ω_i), (ω_0) and ($\omega_i + \omega_0$).*	Boolean	---	---
16	RMS DEVM limit test passed - all packets	Boolean	---	---

5 Remote Operation

17	Peak DEVM limit test passed - all packets	Boolean	---	---
18	Percentage of symbols limit test passed	Boolean	---	---
19	Guard interval limit test passed - all packets	Boolean	---	---
20	Worst case initial frequency error (ω_i).	Real number	Hz	1 Hz
21	Worst case frequency error (ω_0).	Real number	Hz	1 Hz
22	Worst case block frequency error ($\omega_i + \omega_0$)	Real number	Hz	1 Hz
23	Worst case RMS DEVM	Real number	%	0.001 %
24	Worst case peak DEVM	Real number	%	0.001 %
25	% symbols meeting DEVM limit worst case.	Real number	%	0.01 %
26	Worst case guard interval	Real number	s	50 ns
27	Packet number for worst case ω_i	Integer	---	---
28	Packet number for worst case ω_0	Integer	---	---
29	Packet number for worst case ($\omega_i + \omega_0$)	Integer	---	---
30	Packet number for worst case RMS DEVM	Integer	---	---
31	Packet number for worst case peak DEVM	Integer		---
32	Packet number for worst case % symbols meeting limit	Integer		---
33	Packet number for worst case guard interval	Integer		---

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:BT:ECM? 1,1
```

:FETCh:SRWireless:SUMMary:BT:EDPencoding?

Function

Fetch the *Bluetooth* EDR differential phase encoding measurements (TRM/CA/12/C).

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:EDPencoding?
```

Response

#	Description	Format	Units
1	Segment index	Integer	---
2	Packet index (always zero)	Integer	---
3	Extension code (currently zero)	Integer	---
4	Bit error rate (BER) calculated over all packets	Real Number	%
5	Number of bit errors detected over all packets	Integer	---
6	Total number of bits tested over all packets	Integer	---
7	Packet error rate (PER)	Real Number	%
8	PER limit test passed	Boolean	---
9	Number of packets measured	Integer	---
10	Number of bit errors in worst case packet	Integer	---
11	Packet number of worst case packet	Integer	dBm

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code (currently zero)	Number

Example of Use

```
:FETC:SRW:SUMM:BT:EDP? 1,0
```

5

Remote Operation

:FETCh:SRWireless:SUMMary:BT:ERTPower?

Function

Fetch *Bluetooth* measurements - EDR relative TX power (TRM/CA/10/C).

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:ERTPower?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	P_FSK - average for segment	Real number	dBm	0.01 dB
4	P_PSK - average for segment	Real number	dBm	0.01 dB
5	Max power difference: P_PSK - P_FSK	Real number	dB	0.01 dB
6	Relative transmit power limit test passed for all packets	Boolean	---	---
7, 8	Min power difference: P_PSK - P_FSK	Real number	dB	0.01 dB
9	Average power difference (all packets)	Real number	dB	0.01 dB
10	Packet number for max power difference	Integer	---	---
	Packet number for min power difference	Integer	---	---

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:ERTP? 1
```

:FETCh:SRWireless:SUMMary:BT:EYE:ASCii?

Function

Fetch *Bluetooth* measurements - eye diagram (BR / BLE) – ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:EYE:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Symbol length (oversampling factor)	Integer	---	---
4	GFSK data start: offset relative to p0	Integer	---	---
5	GFSK data stop: offset relative to p0	Integer	---	---
6	Number of points	Integer	---	---
7	First frequency deviation value (averaged over segment)	Real number	Hz	1 Hz
…	…			
	Last frequency deviation value (averaged over segment)	Real number	Hz	1 Hz

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:EYE:ASC? 1
```

5

Remote Operation

:FETCh:SRWireless:SUMMary:BT:EYE[:BINary]?

Function

Fetch *Bluetooth* measurements - eye diagram (BR / BLE) – binary format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:EYE[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Segment index	INT32	---
3	Packet index (always zero)	INT32	---
4	Symbol length (oversampling factor)	INT32	---
5	GFSK data start: offset relative to p0	INT32	---
6	GFSK data stop: offset relative to p0	INT32	---
7	Number of points	INT32	---
8	First frequency deviation value (averaged over segment)	REAL32	Hz
…	…		
	Last frequency deviation value (averaged over segment)	REAL32	Hz

Parameters

#	Description	Format
1	Segment index	Number

Note:

1. Refer to sections 5.3.3 and 5.3.4 *Binary_Header_Information* and *Binary_Data_Fields*.

Example of Use

```
:FETC:SRW:SUMM:BT:EYE? 1
```

:FETCh:SRWireless:SUMMary:BT:FRANge?

Function

Fetch Bluetooth frequency range measurements (TRM/CA/04/C)

Currently no extension codes have been defined for this command. Use zero.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:FRANge?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Center frequency	Integer	Hz	1 Hz
5	Peak power	Real number	dBm	0.01 dB
6	Low frequency (f_L) - worst case	Integer	Hz	1 Hz
7	High frequency (f_H) - worst case	Integer	Hz	1 Hz
8	Low frequency limit test passed	Boolean	---	---
9	High frequency limit test passed	Boolean	---	---
10	Spectrum bandwidth	Integer	Hz	1 Hz
11	Number of points	Integer	---	---
12	First power value (averaged over all packets)	Real number	dBm	0.01 dB
…	…			
	Last power value (averaged over all packets)	Real number	dBm	0.01 dB
	Packet number of worst case packet	Integer	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:BT:FRAN? 1,0
```

5

Remote Operation

:FETCh:SRWireless:SUMMary:BT:IBSPurious?

Function

Fetch the *Bluetooth* EDR in band spurious emissions measurements (TRM/CA/13/C).

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:IBSPurious?
```

Response

#	Description	Format	Units
1	Segment index	Integer	---
2	Packet index (always zero)	Integer	---
3	Channel–5 power	Real Number	dBm
4	Channel–4 power	Real Number	dBm
5	Channel–3 power	Real Number	dBm
6	Channel–2 power	Real Number	dBm
7	Channel–1 power	Real Number	dBm
8	Channel power	Real Number	dBm
9	Channel+1 power	Real Number	dBm
10	Channel+2 power	Real Number	dBm
11	Channel+3 power	Real Number	dBm
12	Channel+4 power	Real Number	dBm
13	Channel+5 power	Real Number	dBm
14	Overall test result	Boolean	---
15	Channel–5 test passed	Boolean	---
16	Channel–4 test passed	Boolean	---
17	Channel–3 test passed	Boolean	---
18	Channel–2 test passed	Boolean	---
19	Channel–1 test passed	Boolean	---
20	Channel test passed (always pass)	Boolean	---
21	Channel+1 test passed	Boolean	---
22	Channel+2 test passed	Boolean	---
23	Channel+3 test passed	Boolean	---
24	Channel+4 test passed	Boolean	---
25	Channel+5 test passed	Boolean	---
26	Channel–5 worst case packet number	Integer	---
27	Channel–4 worst case packet number	Integer	---
28	Channel–3 worst case packet number	Integer	---
29	Channel–2 worst case packet number	Integer	---
30	Channel–1 worst case packet number	Integer	---

31	Channel worst case packet number	Integer	---
32	Channel+1 worst case packet number	Integer	---
33	Channel+2 worst case packet number	Integer	---
34	Channel+3 worst case packet number	Integer	---
35	Channel+4 worst case packet number	Integer	---
36	Channel+5 worst case packet number	Integer	---

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:IBSP? 1
```

:FETCh:SRWireless:SUMMary:BT:ICFTolerance?

Function

Fetch *Bluetooth* measurements - initial carrier frequency tolerance (TRM/CA/08/C).

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:ICFTolerance?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Average initial frequency tolerance	Real number	Hz	1 Hz
4	Initial carrier frequency tolerance limit test passed	Boolean	---	---
5	Worst case initial frequency tolerance	Real number	Hz	1 Hz
6	Packet number of worst case.	Integer	---	---

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:ICFT? 1
```

:FETCh:SRWireless:SUMMary:BT:MODulation?

Function

Fetch *Bluetooth* measurements - modulation characteristics (TRM/CA/07/C).

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:MODulation?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Average of Δf2 avg	Real Number	Hz	100 Hz
4	Δf2 max worst case	Real Number	Hz	100 Hz
5	Average of Δf1 avg	Real Number	Hz	100 Hz
6	Δf1 max worst case	Real Number	Hz	100 Hz
7	(Δf2 avg / Δf1 avg) worst case	Real Number	---	0.01
8	%Δf2 max > 115 kHz worst case	Real Number	%	0.01 %
9	Δf1 avg lower limit test passed	Boolean	---	---
10	Δf1 avg upper limit test passed	Boolean	---	---
11	Δf2 avg / Δf1 avg limit test passed	Boolean	---	---
12	%Δf2 max > 115 kHz limit test passed	Boolean	---	---
13	Packet number for Δf2 max worst case	Integer	---	---
14	Packet number for Δf1 max worst case	Integer	---	---
15	Packet number for (Δf2 avg / Δf1 avg) worst case	Integer	---	---
16	Packet number for %Δf2 max > 115 kHz worst case	Integer	---	---

Notes on SIG Compliant and Speed Test modes (see :CONFigure:SRWireless:BT:MODE):

1. If the mode is SPEED, all response fields will be populated with results.
2. If the mode is STANDARD, some response fields will be populated with results depending on whether the measurement can be made with the specified packet payload according to the SIG Specification.
3. If a measurement cannot be made in STANDARD mode, the measurement field will contain the value NAN, 'not a number' (defined as 9.91E+37), any associated Boolean flags will be set to 0 (false), and any associated packet number (for example 'Packet number for Δf2 max worst case') will be set to zero.

5 Remote Operation

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:MOD? 1
```

:FETCh:SRWireless:SUMMary:BT:POWer?

Function

Fetch *Bluetooth* measurements - transmit power (TRM/CA/01/C).

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:POWer?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Average of packet average powers	Real number	dBm	0.01 dB
4	Maximum peak power for segment	Real number	dBm	0.01 dB
5	Average power lower limit test passed	Boolean	---	
6	Average power upper limit test passed	Boolean	---	
7	Peak power limit test passed for all packets	Boolean	---	
8	Minimum peak power	Real number	dBm	0.01 dB
9	Packet number containing maximum peak power	Integer	---	
10	Packet number containing minimum peak power	Integer	---	

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:POW? 1
```

:FETCh:SRWireless:SUMMary:BT:PPRofile:AVERage:ASCii?

Function

Fetch *Bluetooth* measurements - power burst profile (BR / EDR) – average data – ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:PPRofile:AVERage:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	
2	Packet index (always zero)	Integer	---	
3	Start time (relative to p0)	Real number	s	1 µs
4	Stop time (relative to p0)	Real number	s	1 µs
5	Number of points (n)	Integer	---	
6	First average power profile point	Real number	dBm	0.01 dB
---	…			
	Last average power profile point	Real number	dBm	0.01 dB
	Index fields valid	Boolean	---	---
	Start offset (always zero)	Integer	---	---
	Stop offset (always zero)	Integer	---	---
	Preamble index (p0)	Integer	---	---
	Sync word index	Integer	---	---
	Trailer index	Integer	---	---
	Packet header index	Integer	---	---
	Guard time index	Integer	---	---
	EDR sync index	Integer	---	---
	Payload header index	Integer	---	---
	Payload index	Integer	---	---
	CRC start index	Integer	---	---
	CRC stop index	Integer	---	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile.　A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:PPR:AVER:ASC? 1
```

:FETCh:SRWireless:SUMMary:BT:PPRofile:AVERage[:BINary]?

Function

Fetch *Bluetooth* measurements - power burst profile (BR / EDR) – average data – binary format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:PPRofile:AVERage[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Segment index	INT32	---
3	Packet index (always zero)	INT32	---
4	Start time (relative to p0)	REAL32	s
5	Stop time (relative to p0)	REAL32	s
6	Number of points (n)	INT32	---
7	First average power profile point	REAL32	dBm
…	…		
	Last average power profile point	REAL32	dBm
	Index fields (valid if non-zero)	INT32	---
	Start offset (always zero)	INT32	---
	Stop offset (always zero)	INT32	---
	Preamble index (p0)	INT32	---
	Sync word index	INT32	---
	Trailer index	INT32	---
	Packet header index	INT32	---
	Guard time index	INT32	---
	EDR sync index	INT32	---
	Payload header index	INT32	---
	Payload index	INT32	---
	CRC start index	INT32	---
	CRC stop index	INT32	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile. A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.

3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.
4. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
5. Binary data will always be a multiple of 4 bytes.
6. Refer to sections 5.3.3 and 5.3.4 *Binary_Header_Information* and *Binary_Data_Fields*.

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:PPR:AVER? 1
```

:FETCh:SRWireless:SUMMary:BT:PPRofile:PEAK:ASCii?

Function

Fetch *Bluetooth* measurements - power burst profile (BR / EDR) – peak data – ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:PPRofile:PEAk:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	
2	Packet index (always zero)	Integer	---	
3	Start time (relative to p0)	Real number	s	1 μs
4	Stop time (relative to p0)	Real number	s	1 μs
5	Number of points (n)	Integer	---	
6	First peak power profile point	Real number	dBm	0.01 dB
---	...			
	Last peak power profile point	Real number	dBm	0.01 dB
	Index fields valid	Boolean	---	---
	Start offset (always zero)	Integer	---	---
	Stop offset (always zero)	Integer	---	---
	Preamble index (p0)	Integer	---	---
	Sync word index	Integer	---	---
	Trailer index	Integer	---	---
	Packet header index	Integer	---	---
	Guard time index	Integer	---	---
	EDR sync index	Integer	---	---
	Payload header index	Integer	---	---
	Payload index	Integer	---	---
	CRC start index	Integer	---	---
	CRC stop index	Integer	---	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile. A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.

5

Remote Operation

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:PPR:PEAK:ASC? 1
```

:FETCh:SRWireless:SUMMary:BT:PProfile:PEAK[:BINary]?

Function

Fetch *Bluetooth* measurements - power burst profile (BR / EDR) – peak data – binary format.

Command

N/A

Query

`:FETCh:SRWireless:SUMMary:BT:PPRofile:PEAK[:BINary]?`

Response

#	Description	Format	Units
1	#nmm…mm	Binary header	---
2	Segment index	INT32	---
3	Packet index (always zero)	INT32	---
4	Start time (relative to p0)	REAL32	s
5	Stop time (relative to p0)	REAL32	s
6	Number of points (n)	INT32	---
7	First peak power profile point	REAL32	dBm
…	…		
	Last peak power profile point	REAL32	dBm
	Index fields (valid if non-zero)	INT32	---
	Start offset (always zero)	INT32	---
	Stop offset (always zero)	INT32	---
	Preamble index (p0)	INT32	---
	Sync word index	INT32	---
	Trailer index	INT32	---
	Packet header index	INT32	---
	Guard time index	INT32	---
	EDR sync index	INT32	---
	Payload header index	INT32	---
	Payload index	INT32	---
	CRC start index	INT32	---
	CRC stop index	INT32	---

Notes:

1. The index fields contain offsets from the start of the power profile array that identify the positions of specific fields within the power burst profile. A typical use is for plotting a power burst profile with each field displayed in a different color.
2. The 'index field valid' flag is set to 1 (true) if the index fields contain valid data.
3. The 'start offset' and 'stop offset' fields are reserved for future use. They always contain zero.

4. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
5. Binary data will always be a multiple of 4 bytes.
6. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:PPR:PEAK? 1
```

:FETCh:SRWireless:SUMMary:BT:SPECtrum:ASCii?

Function

Fetch *Bluetooth* measurements - spectrum (BR / EDR / BLE) – ASCII format

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:BT:SPECtrum:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Bandwidth (10 MHz for *Bluetooth*)	Real number	Hz	1 Hz
4	Number of points in spectrum	Integer	---	---
5	First power value (averaged over segment)	Real number	dBm	0.01 dB
	…			
	Last power value (averaged over segment)	Real number	dBm	0.01 dB

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:BT:SPEC:ASC? 1
```

5

Remote Operation

:FETCh:SRWireless:SUMMary:BT:SPECtrum[:BINary]?

Function

Fetch *Bluetooth* measurements - spectrum (BR / EDR / BLE) – binary format

Command

N/A

Query

`:FETCh:SRWireless:SUMMary:BT:SPECtrum[:BINary]?`

Response

#	Description	Format	Units	Resolution
1	#nmmm… mm	Binary header	---	---
2	Segment index	INT32	---	---
3	Packet index (always zero)	INT32	---	---
4	Bandwidth	REAL32	Hz	1 Hz
5	Number of points	INT32	---	---
6	First power value (averaged over segment)	REAL32	dBm	0.01 dB
…	…			
	Last power value (averaged over segment)	REAL32	dBm	0.01 dB

Parameters

#	Description	Format
1	Segment index	Number

Note:

1. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

Example of Use

`:FETC:SRW:SUMM:BT:SPEC? 1`

:FETCh:SRWireless:SUMMary:CW:FREQuency?

Function

Fetch CW frequency offset. This is the offset between the frequency generated by the device under test and the frequency to which the instrument receiver is tuned. (The receiver frequency for the active segment can be read using :CONF:SRW:FREQ?)

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:CW:FREQuency?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Frequency offset	Real Number	Hz	1 Hz
4	Frequency limit test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:CW:FREQ? 1

> 1,0,123000,1
```

:FETCh:SRWireless:SUMMary:CW:FREQuency? (Extended)

Function

Fetch CW frequency with optional extension code.
If the extension code is zero, the frequency offset relative to the VSA frequency is returned.
If bit 0 of the extension code is set, both the absolute frequency and the offset are returned.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:CW:FREQuency?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code	Integer	---	---
4	Frequency offset from VSA frequency.	Integer	Hz	1 Hz
	If bit 0 of extension code set:			
5	Frequency	Integer	Hz	1 Hz
6	Frequency offset limit test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:CW:FREQ? 1,1

> 1,0,1,123000,2412123000,1
```

:FETCh:SRWireless:SUMMary:CW:POWer?

Function

Fetch CW power (average and peak)

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:CW:POWer?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Average power over segment.	Real number	dBm	0.01 dB
4	Peak power recorded in segment.	Real number	dBm	0.01 dB
5	Average power upper limit test passed.	Boolean	---	---
6	Average power lower limit test passed.	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:CW:POW? 1

> 1,0,-10.12,-9.15,1,1
```

:FETCh:SRWireless:SUMMary:CW:SPECtrum:ASCii?

Function

Fetch CW spectrum – ASCII format

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:CW:SPECtrum:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Number of points in spectrum	Integer	---	---
4	First power value - averaged over segment	Real number	dBm	0.01 dB
…				
n	Last power value - averaged over segment	Real number	dBm	0.01 dB

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:CW:SPEC:ASC? 1

> 1,0,4096,-55.12,…,-61.23
```

:FETCh:SRWireless:SUMMary:CW:SPECtrum[:BINary]?

Function

Fetch CW spectrum – binary format

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:CW:SPECtrum[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary header.	---
2	Segment index	INT32	---
3	Packet index (always zero)	INT32	---
4	Number of points	INT32	---
5	First power value	REAL32	dBm
…			
	Last power value	REAL32	dBm

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:CW:SPEC? 1
```

Notes:

1. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
2. Binary data will always be a multiple of 4 bytes.
3. Refer to sections 5.3.3 and 5.3.4 Binary_Header_Information and Binary_Data_Fields.

:FETCh:SRWireless:SUMMary:WLAN:CCDF[:ASCii]?

Function

Fetch WLAN CCDF profile (graphical) measurements - ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:CCDF:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Number of gates	Integer	---	---
	For first gate			
1	Gate number	Integer	---	---
2	Number of points in CCDF profile	Integer	---	---
3	First CCDF value (accumulated over segment)	Real number	%	0.001%
…	…			
	Last CCDF value (accumulated over segment)	Real number	%	0.001%
	…			
	For last gate			
1	Gate number	Integer	---	---
2	Number of points in CCDF profile	Integer	---	---
3	First CCDF value (accumulated over segment)	Real number	%	0.001%
…	…			
	Last CCDF value (accumulated over segment)	Real number	%	0.001%

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:CCDF:ASC? 1
```

:FETCh:SRWireless:SUMMary:WLAN:CCDF[:BINary]?

Function

Fetch WLAN CCDF profile (graphical) measurements - binary format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:CCDF[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm...mm	Binary header	---
2	Segment index	INT32	---
3	Packet index (always zero)	INT32	---
4	Number of gates	INT32	---
	For first gate:		
1	Gate number	INT32	---
2	Number of points	INT32	---
3	First CCDF value	REAL32	%
…	…		
	Last CCDF value	REAL32	%
…			
	For last gate:		
1	Gate number	INT32	---
2	Number of points	INT32	---
3	First CCDF value	REAL32	%
…	…		
	Last CCDF value	REAL32	%

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:CCDF? 1
```

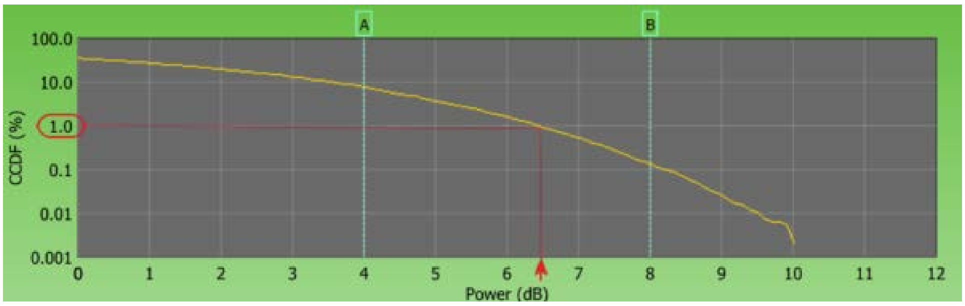
Notes:

1. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
2. Binary data will always be a multiple of 4 bytes.
3. Refer to sections 5.3.3 and 5.3.4 Binary Header Information and Binary Data Fields.

:FETCh:SRWireless:SUMMary:WLAN:CCDF:PDISt?

Function

Fetch CCDF power distribution spot measurement, corresponding to the 'percentage of samples' value set using `:CONFigure:SRWireless:WLAN:CCDF:PSAMples`.



For the example CCDF measurement shown above, `:CONFigure:SRWireless:WLAN:CCDF:PSAMples` has been set to 1.0%. `:FETC:SRW:SUMM:WLAN:CCDF:PDIS?` will return the power distribution value corresponding to the point on the x-axis where the CCDF curve crosses the 1.0% line. In this case approximately 6.5 dB.

This means that the signal power exceeded the average power by at least 6.5 dB in 1% of the measured samples - (equivalent to 1% of the time)

Command

N/A

Query

`:FETCh:SRWireless:SUMMary:WLAN:CCDF:PDISt?`

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code (reserved – always zero)	Integer	---	---
4	Percent samples	Real number	%	0.001%
5	Number of gates	Integer	---	---
	Results for first gate:			
1	Gate number	Integer	---	---
2	Power distribution (for segment)	Real number	dB	0.01 dB
…	…			
	Results for last gate:			
1	Gate number	Integer	---	---

2	Power distribution (for segment)	Real number	dB	0.01 dB

Parameters

#	Description	Format	Units	Min	Max	Def	Resolution
1	Segment index	Number	---	---	---	---	---
2	Extension code	Number	---	0	$2^{32}-1$	0	---

Example of Use

```
:FETC:SRW:SUMM:WLAN:CCDF:PDIS? 1,0
```

:FETCh:SRWireless:SUMMary:WLAN:CPOWer?

Function

Fetch cross power measurements.
Returns the power of each spatial stream of a T-MIMO or S-MIMO signal and indicates the dominant stream.

Currently no extension codes have been defined for this command. Use zero.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:CPOWer?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Dominant spatial stream number	Integer	---	---
5	Number of spatial streams	Integer	---	---
	For each spatial stream:	Integer	---	---
1	Average power (averaged over segment)	Real number	dBm	0.01 dB
2	Maximum average power (from packet with highest average power)	Real number	dBm	0.01 dB
3	Minimum average power (from packet with lowest average power)	Real number	dBm	0.01 dB
4	Packet number of packet with highest average power	Integer	---	---
5	Packet number of packet with lowest average power	Integer	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:CPOW? 1,0
```

:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM? (DEPRECATED)

Function

Fetch basic WLAN EVM measurements - DSSS packets.

Note:

This command is now deprecated. Please see the following commands, which clearly differentiate between results in percent and dB units:

```
:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:PCT?
:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:PCT?
:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:DB?
:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:DB?
```

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	DSSS EVM max - dB	Real number	dB	0.01 dB
4	DSSS EVM max - %	Real number	%	0.1 %
5	DSSS EVM min - dB	Real number	dB	0.01 dB
6	DSSS EVM min - %	Real number	%	0.1 %
7	DSSS EVM avg - dB	Real number	dB	0.01 dB
8	DSSS EVM avg %	Real number	%	0.1 %
9	DSSS center freq tolerance - ppm	Real number	ppm	0.1 ppm
10	DSSS center freq tolerance Hz	Real number	Hz	1 Hz
11	DSSS phase error	Real number	degrees	0.01 degrees
12	DSSS magnitude error	Real number	---	0.01
13	DSSS IQ offset	Real number	---	0.01
14	DSSS chip clock error - ppm	Real number	ppm	0.1 ppm
15	DSSS chip clock error Hz	Real number	Hz	1 Hz
16	DSSS EVM test passed	Boolean	---	---
17	DSSS center freq tolerance test passed	Boolean	---	---
18	DSSS chip clock test passed	Boolean	---	---
19	Packet number for max EVM	Integer	---	---
20	Packet number for min EVM	Integer	---	---

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:DSSS:EVM? 1
```

:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM? (Extended) (DEPRECATED)

Function

Fetch WLAN EVM measurements - DSSS packets - using extension code.

Note:

This command is now deprecated. Please see the following commands, which clearly differentiate between results in percent and dB units:

```
:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:PCT?
:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:PCT?
:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:DB?
:FETCh:SRWireless:PACKet:WLAN:DSSS:EVM:DB?
```

The optional extension code adds additional results to the response as described in the following table:

Extension code bit	Additional fields in response
0	DSSS carrier suppression from IQ offset
1	Amplitude and phase imbalance.
2	EVM (peak) max - highest symbol EVM from all packets

Note:

The EVM limit check is performed on the EVM (peak) max measurement

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code	Integer	---	---
4	DSSS EVM (RMS) max - dB (from packet whose average EVM is highest)	Real number	dB	0.01 dB
5	DSSS EVM (RMS) max - % (from packet whose average EVM is highest)	Real number	%	0.1 %
6	DSSS EVM (RMS) min - dB (from packet whose average EVM is lowest)	Real number	dB	0.01 dB
7	DSSS EVM (RMS) min - % (from packet whose average EVM is lowest)	Real number	%	0.1 %

8	DSSS EVM (RMS) avg - dB (averaged over segment)	Real number	dB	0.01 dB
9	DSSS EVM (RMS) avg % (averaged over segment)	Real number	%	0.1 %
	If bit 2 of Extension Code is set:			
10	DSSS EVM (peak) max - dB (highest symbol EVM from all packets)	Real number	dB	0.01 dB
11	DSSS EVM (peak) max - % (highest symbol EVM from all packets. Used for limit check)	Real number	%	0.1 %
12	DSSS center freq tolerance - ppm	Real number	ppm	0.1 ppm
13	DSSS center freq tolerance Hz	Real number	Hz	1 Hz
14	DSSS phase error	Real number	degrees	0.01 degrees
15	DSSS magnitude error	Real number	%	0.01
16	DSSS IQ offset	Real number	---	0.01
	If bit 0 of Extension Code is set:			
17	DSSS carrier suppression from IQ offset (averaged over segment)	Real number	dB	0.01
18	DSSS chip clock error - ppm	Real number	ppm	0.1 ppm
19	DSSS chip clock error Hz	Real number	Hz	1 Hz
	If bit 1 of Extension Code is set:			
20	DSSS amplitude imbalance (averaged over segment)	Real number	dB	0.01 dB
21	DSSS phase imbalance (averaged over segment)	Real number	degrees	0.01 degrees
22	DSSS EVM test passed	Boolean	---	---
23	DSSS center freq tolerance test passed	Boolean	---	---
24	DSSS chip clock test passed	Boolean	---	---
	If bit 0 of Extension Code is set:			
25	DSSS carrier suppression from IQ offset test passed.	Boolean	---	---
26	Packet number for EVM (RMS) max	Integer	---	---
27	Packet number for EVM (RMS) min	Integer	---	---
	If bit 2 of Extension Code is set:			
28	Packet number for EVM (peak) max.	Integer	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:DSSS:EVM? 1,1   // Bit 0 of extension code is set
```

:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:DB?

Function

Fetch WLAN EVM measurements in dB units.

This command replaces `:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM?` which has been marked as deprecated.

(See also `:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:PCT?`)

The extension code adds additional results to the response as described in the following table:

Extension code bit	Additional fields in response
0	DSSS carrier suppression from IQ offset
1	Amplitude and phase imbalance.
2	EVM (peak) max - highest symbol EVM from all packets

Notes:
The EVM limit check is performed on the EVM (peak) max measurement.

The IQ imbalance measurements (phase imbalance and amplitude imbalance) are only applicable to signals where the modulation type is QPSK. If there is no QPSK component in the signal to be measured, the values 0.0 degrees for phase imbalance and 0.0 dB for amplitude imbalance are returned.

IQ imbalance measurements are not applicable for the following cases:

1. Data rate of 1 Mbps. (Uses BPSK modulation only.)
2. Any DSSS packet type with a long preamble and long header (IEEE Long PPDU format) where the EVM calculation method is set to "PREAMBLE_HDR".
   The reason is that selecting "PREAMBLE_HDR" limits the analysis to the preamble and header only, and for "long preamble and header" packets the modulation type is BPSK only.

Command

N/A

Query

`:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:DB?`

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---

3	Extension code	Integer	---	---
4	DSSS EVM (RMS) max - dB (from packet whose average EVM is highest)	Real number	dB	0.01 dB
5	DSSS EVM (RMS) min - dB (from packet whose average EVM is lowest)	Real number	dB	0.01 dB
6	DSSS EVM (RMS) avg dB (averaged over segment)	Real number	dB	0.01 dB
	If bit 2 of Extension Code is set:			
7	DSSS EVM (peak) max - dB (highest symbol EVM from all packets. Used for limit check)	Real number	dB	0.01 dB
8	DSSS center freq tolerance - ppm	Real number	ppm	0.1 ppm
9	DSSS center freq tolerance Hz	Real number	Hz	1 Hz
10	DSSS phase error	Real number	degrees	0.01 degrees
11	DSSS magnitude error	Real number	%	0.01
12	DSSS IQ offset	Real number	---	0.01
	If bit 0 of Extension Code is set:			
13	DSSS carrier suppression from IQ offset (averaged over segment)	Real number	dB	0.01
14	DSSS chip clock error - ppm	Real number	ppm	0.1 ppm
15	DSSS chip clock error Hz	Real number	Hz	1 Hz
	If bit 1 of Extension Code is set:			
16	DSSS amplitude imbalance (averaged over segment)	Real number	dB	0.01 dB
17	DSSS phase imbalance (averaged over segment)	Real number	degrees	0.01 degrees
18	DSSS EVM test passed	Boolean	---	---
19	DSSS center freq tolerance test passed	Boolean	---	---
20	DSSS chip clock test passed	Boolean	---	---
	If bit 0 of Extension Code is set:			
21	DSSS carrier suppression from IQ offset test passed.	Boolean	---	---
22	Packet number for EVM (RMS) max	Integer	---	---
23	Packet number for EVM (RMS) min	Integer	---	---
	If bit 2 of Extension Code is set:			
24	Packet number for EVM (peak) max.	Integer	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:DSSS:EVM:DB? 1,1        // Bit 0 of extension code is set
```

:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:PCT?

Function

Fetch WLAN EVM measurements in units of percent.

This command replaces `:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM?` which has been marked as deprecated.

(See also `:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:DB?`)

The extension code adds additional results to the response as described in the following table:

Extension code bit	Additional fields in response
0	DSSS carrier suppression from IQ offset
1	Amplitude and phase imbalance.
2	EVM (peak) max - highest symbol EVM from all packets

Notes:

The EVM limit check is performed on the EVM (peak) max measurement

The IQ imbalance measurements (phase imbalance and amplitude imbalance) are only applicable to signals where the modulation type is QPSK. If there is no QPSK component in the signal to be measured, the values 0.0 degrees for phase imbalance and 0.0 dB for amplitude imbalance are returned.

IQ imbalance measurements are not applicable for the following cases:

1. Data rate of 1 Mbps. (Uses BPSK modulation only.)
2. Any DSSS packet type with a long preamble and long header (IEEE Long PPDU format) where the EVM calculation method is set to "PREAMBLE_HDR".
   The reason is that selecting "PREAMBLE_HDR" limits the analysis to the preamble and header only, and for "long preamble and header" packets the modulation type is BPSK only.

Command

N/A

Query

`:FETCh:SRWireless:SUMMary:WLAN:DSSS:EVM:PCT?`

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---

3	Extension code	Integer	---	---
4	DSSS EVM (RMS) max - % (from packet whose average EVM is highest)	Real number	%	0.1 %
5	DSSS EVM (RMS) min - % (from packet whose average EVM is lowest)	Real number	%	0.1 %
6	DSSS EVM (RMS) avg % (averaged over segment)	Real number	%	0.1 %
	If bit 2 of Extension Code is set:			
7	DSSS EVM (peak) max - % (highest symbol EVM from all packets. Used for limit check)	Real number	%	0.1 %
8	DSSS center freq tolerance - ppm	Real number	ppm	0.1 ppm
9	DSSS center freq tolerance Hz	Real number	Hz	1 Hz
10	DSSS phase error	Real number	degrees	0.01 degrees
11	DSSS magnitude error	Real number	%	0.01
12	DSSS IQ offset	Real number	---	0.01
	If bit 0 of Extension Code is set:			
13	DSSS carrier suppression from IQ offset (averaged over segment)	Real number	dB	0.01
14	DSSS chip clock error - ppm	Real number	ppm	0.1 ppm
15	DSSS chip clock error Hz	Real number	Hz	1 Hz
	If bit 1 of Extension Code is set:			
16	DSSS amplitude imbalance (averaged over segment)	Real number	dB	0.01 dB
17	DSSS phase imbalance (averaged over segment)	Real number	degrees	0.01 degrees
18	DSSS EVM test passed	Boolean	---	---
19	DSSS center freq tolerance test passed	Boolean	---	---
20	DSSS chip clock test passed	Boolean	---	---
	If bit 0 of Extension Code is set:			
21	DSSS carrier suppression from IQ offset test passed.	Boolean	---	---
22	Packet number for EVM (RMS) max	Integer	---	---
23	Packet number for EVM (RMS) min	Integer	---	---
	If bit 2 of Extension Code is set:			
24	Packet number for EVM (peak) max.	Integer	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:DSSS:EVM:PCT? 1,1       // Bit 0 of extension code is set
```

:FETCh:SRWireless:SUMMary:WLAN:DSSS:PRAMps?

Function

Fetch WLAN power profile (numeric) measurements - power up and down ramps – DSSS packets

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:DSSS:PRAMps?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Average power-on ramp time for segment	Real number	s	0.01 µs
4	Average power-down ramp time for segment	Real number	s	0.01 µs
5	Power-on ramp time test passed	Boolean	---	---
6	Power-down ramp time test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:DSSS:PRAM? 1
```

:FETCh:SRWireless:SUMMary:WLAN:OFDM:AEVM?

Function

Fetch WLAN advanced EVM measurements for OFDM packets.

Command

N/A

Query

`:FETCh:SRWireless:SUMMary:WLAN:OFDM:AEVM?`

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer		
2	Packet index (always zero)	Integer		
3	Number of symbols	Integer		
4	Number of data subcarriers	Integer		
5	Number of pilot subcarriers	Integer		
6	Number of subcarriers	Integer		
7	First element of spectral flatness array (averaged over segment)	Real number	dB	0.01 dB
	…			
	Last element of spectral flatness array (averaged over segment)	Real number	dB	0.01 dB
	First element of EVM per data subcarrier array (averaged over segment)	Real number	dB	0.1 dB
	…			
	Last element of EVM per data subcarrier array (averaged over segment)	Real number	dB	0.1 dB
	First element of EVM per pilot subcarrier array (averaged over segment)	Real number	dB	0.1 dB
	…			
	Last element of EVM per pilot subcarrier array (averaged over segment)	Real number	dB	0.1 dB
	First element of EVM per symbol array (averaged over segment)	Real number	dB	0.1 dB
	…			
	Last element of EVM per symbol array (averaged over segment)	Real number	dB	0.1 dB

Notes:

1. Length of spectral flatness array is equal to the number of subcarriers (field #6).
2. Length of EVM per data subcarrier array is equal to the number of data subcarriers (field #4).
3. Length of EVM per pilot subcarrier array is equal to the number of pilot subcarriers (field #5).
4. Length of EVM per symbol array is equal to the number of symbols (field #3).

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:OFDM:AEVM? 1
```

:FETCh:SRWireless:SUMMary:WLAN:OFDM:CESTimation?

Function

Fetch the channel estimation values (amplitude and phase).

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:OFDM:CESTimation?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Number of spatial streams	Integer	---	---
5	Number of values each of amplitude and phase results	Integer	---	---
6	First amplitude value (averaged over segment)	Real number	V / Hz	0.000001
	…	---	---	---
	Last amplitude value (averaged over segment)	Real number	V / Hz	0.000001
	First phase value (averaged over segment)	Real number	radians	0.000001
	…	---	---	---
	Last phase value (averaged over segment)	Real number	radians	0.000001

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number (currently zero)

Example of Use

```
:FETC:SRW:SUMM:WLAN:OFDM:CEST? 1,0
```

:FETCh:SRWireless:SUMMary:WLAN:OFDM:CESTimation:AMPLitude?

Function

Fetch the channel estimation values (amplitude only).

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:OFDM:CESTimation:AMPLitude?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Number of spatial streams	Integer	---	---
5	Number of amplitude values	Integer	---	---
6	First amplitude value (averaged over segment)	Real number	V / Hz	0.000001
	…	---	---	---
	Last amplitude value (averaged over segment)	Real number	V / Hz	0.000001

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number (currently zero)

Example of Use

```
:FETC:SRW:SUMM:WLAN:OFDM:CEST:AMPL? 1,0
```

5

Remote Operation

:FETCh:SRWireless:SUMMary:WLAN:OFDM:CESTimation:PHASe?

Function

Fetch the channel estimation values (phase only).

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:OFDM:CESTimation:PHASe?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Number of spatial streams	Integer	---	---
5	Number of phase values	Integer	---	---
6	First phase value (averaged over segment)	Real number	radians	0.000001
	…	---	---	---
	Last phase value (averaged over segment)	Real number	radians	0.000001

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number (currently zero)

Example of Use

```
:FETC:SRW:SUMM:WLAN:OFDM:CEST:PHAS? 1,0
```

:FETCh:SRWireless:SUMMary:WLAN:OFDM:EVM?

Function

Fetch WLAN basic EVM measurements - OFDM packets.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:OFDM:EVM?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer		
2	Packet index (always zero)	Integer		
3	OFDM number of subcarriers	Integer		
4	OFDM number of symbols	Integer		
5	OFDM center frequency leakage (averaged over segment)	Real number	dB	0.01 dB
6	OFDM symbol clk freq tolerance ppm (averaged over segment)	Real number	ppm	0.1 ppm
7	OFDM center freq tolerance ppm (averaged over segment)	Real number	ppm	0.1 ppm
8	OFDM center freq tolerance Hz (averaged over segment)	Real number	Hz	1 Hz
9	OFDM EVM max - dB (from packet whose average EVM is highest)	Real number	dB	0.01 dB
10	OFDM EVM max - % (from packet whose average EVM is highest)	Real number	%	0.01 %
11	OFDM EVM min - dB (from packet whose average EVM is lowest)	Real number	dB	0.01 dB
12	OFDM EVM min - % (from packet whose average EVM is lowest)	Real number	%	0.01 %
13	OFDM EVM avg - dB (averaged over segment)	Real number	dB	0.01 dB
14	OFDM EVM avg % (averaged over segment)	Real number	%	0.01 %
15	OFDM center frequency leakage test passed	Boolean	---	---
16	OFDM symbol clk freq tolerance test passed	Boolean	---	---
17	OFDM center freq tolerance test passed	Boolean	---	---
18	OFDM EVM test passed	Boolean	---	---
19	OFDM spectral flatness test passed	Boolean	---	---
20	Packet number for max EVM	Integer	---	---

21	Packet number for min EVM	Integer	---	---

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:OFDM:EVM? 1
```

:FETCh:SRWireless:SUMMary:WLAN:OFDM:EVM? (Extended)

Function

Fetch WLAN basic EVM measurements - OFDM packets - using extension code.

The optional extension code adds additional results to the response as described in the following table:

Extension code bit	Additional fields in response
0	IQ imbalance and phase error measurements.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:OFDM:EVM?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer		
2	Packet index (always zero)	Integer		
3	Extension code	Integer		
4	OFDM number of subcarriers	Integer		
5	OFDM number of symbols	Integer		
6	OFDM center frequency leakage (averaged over segment)	Real number	dB	0.01 dB
7	OFDM symbol clk freq tolerance ppm (averaged over segment)	Real number	ppm	0.1 ppm
8	OFDM center freq tolerance ppm (averaged over segment)	Real number	ppm	0.1 ppm
9	OFDM center freq tolerance Hz (averaged over segment)	Real number	Hz	1 Hz
10	OFDM EVM max - dB (from packet whose average EVM is highest)	Real number	dB	0.01 dB
11	OFDM EVM max - % (from packet whose average EVM is highest)	Real number	%	0.01 %
12	OFDM EVM min - dB (from packet whose average EVM is lowest)	Real number	dB	0.01 dB
13	OFDM EVM min - % (from packet whose average EVM is lowest)	Real number	%	0.01 %
14	OFDM EVM avg - dB (averaged over segment)	Real number	dB	0.01 dB
15	OFDM EVM avg % (averaged over segment)	Real number	%	0.01 %

	If bit 0 of extension code set:			
16	OFDM amplitude imbalance (averaged over segment)	Real number	dB	0.01 dB
17	OFDM phase imbalance (averaged over segment)	Real number	degrees	0.01°
18	OFDM phase error	Real number	degrees	0.01°
19	OFDM center frequency leakage test passed	Boolean	---	---
20	OFDM symbol clk freq tolerance test passed	Boolean	---	---
21	OFDM center freq tolerance test passed	Boolean	---	---
22	OFDM EVM test passed	Boolean	---	---
23	OFDM spectral flatness test passed	Boolean	---	---
24	Packet number for max EVM	Integer	---	---
25	Packet number for min EVM	Integer	---	---

Note:

Amplitude imbalance, phase imbalance and phase error are not available when the wireless standard defined for the segment is composite MIMO (COMPMIMO). Not a number (NaN) values are returned for these measurements in that case.

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:OFDM:EVM? 1, 1
```

:FETCh:SRWireless:SUMMary:WLAN:OFDM:EVM:COMBined?

Function

Fetch combined WLAN EVM measurements.
Combined EVM measurements are available for 802.11ac 80+80 signals, where a combined EVM of the upper and lower frequency segments is calculated.

There are currently no extension codes defined for this command.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:OFDM:EVM:COMBined?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	OFDM combined EVM max – dB (from packet whose combined EVM is highest)	Real number	dB	0.1 dB
5	OFDM combined EVM max – % (from packet whose combined EVM is highest)	Real number	%	0.1%
6	OFDM combined EVM min – dB (from packet whose combined EVM is lowest)	Real number	dB	0.1 dB
7	OFDM combined EVM min – % (from packet whose combined EVM is lowest)	Real number	%	0.1%
8	OFDM combined EVM avg – dB (averaged over segment)	Real number	dB	0.1 dB
9	OFDM combined EVM avg – % (averaged over segment)	Real number	%	0.1%
10	OFDM combined EVM test passed	Boolean		

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:OFDM:EVM:COMB? 1,0
```

5

Remote Operation

:FETCh:SRWireless:SUMMary:WLAN:OFDM:FERRor?

Function

Fetch OFDM preamble frequency error v time measurement.

Currently no extension codes have been defined for this command. Use zero.

Command

N/A

Query

`:FETCh:SRWireless:SUMMary:WLAN:OFDM:FERRor?`

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Timebase length	Integer	μs	1 μs
5	Samples / μs	Integer	---	---
6	Center frequency tolerance (averaged over segment)	Integer	Hz	1 Hz
7	Number of sample periods in timebase length	Integer	---	---
8	Offset (in samples) to start of short training sequence	Integer	---	---
9	Number of samples in short training sequence	Integer	---	---
	For each sample in short training sequence:	Integer	---	---
	Frequency error (averaged over segment)	Integer	Hz	1 Hz
	Offset (in samples) to start of long training sequence	Integer	---	---
	Number of samples in long training sequence	Integer	---	---
	For each sample in long training sequence:	Integer	---	---
	Frequency error (averaged over segment)	Integer	Hz	1 Hz

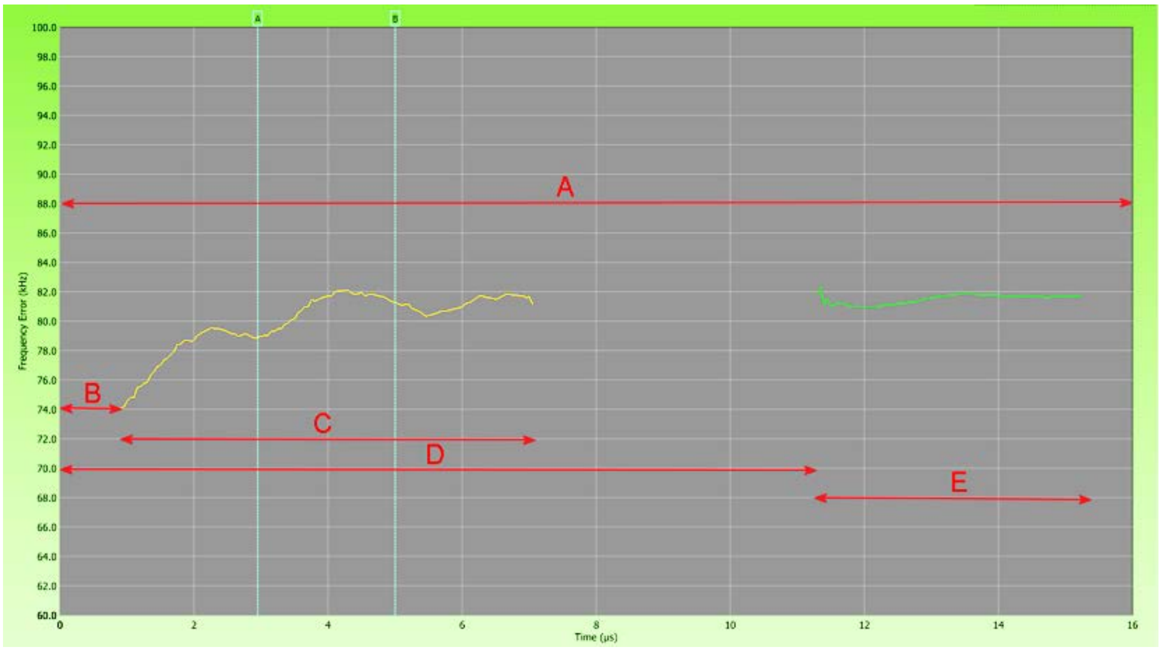
Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

`:FETC:SRW:SUMM:WLAN:OFDM:FERR? 1,0`

Using the Response Fields to Display the Measurement



The graph shows a display of frequency error v time during a packet preamble. The yellow trace shows the short training sequence and the green trace shows the long training sequence plotted on a timebase spanning the packet preamble.

The following table explains how this graph can be constructed from the **response fields** returned by this command.

Timebase Length	See **A**. This is the length of the OFDM packet preamble in microseconds. Within this time frame are plotted the frequency error graphs for two fields in the preamble: the short training sequence and the long training sequence.
Number of sample periods in Timebase Length	This is the total number of points that can be plotted within the Timebase Length. (It is equal to **Timebase Length** multiplied by **Samples / µs**.)
Offset (in samples) to start of short training sequence	See **B**. This value defines where along the x axis to start plotting the short training sequence results.
Number of samples in short training sequence	See **C**. This value defines the length (in samples) of the short training sequence graph.
Offset (in samples) to start of short training sequence	See **D**. This value defines where along the x axis to start plotting the long training sequence results.
Number of samples in long training sequence	See **E**. This value defines the length (in samples) of the long training sequence graph.

:FETCh:SRWireless:SUMMary:WLAN:OFDM:PERRor?

Function

Fetch OFDM phase error v symbol measurement.

Currently no extension codes have been defined for this command. Use zero.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:OFDM:PERRor?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Number of symbols	Integer	---	---
	For each symbol:			
1	Phase error (averaged over segment)	Real number	degrees	0.01°

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:OFDM:PERR? 1,0
```

:FETCh:SRWireless:SUMMary:WLAN:POWer?

Function

Fetch WLAN transmit power (numeric) measurements.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:POWer?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Number of gates	Integer	---	---
	Results for first gate:			
1	Gate number	Integer	---	---
2	Peak power recorded in segment	Real number	dBm	0.01 dB
3	Average power of all packets in segment	Real number	dBm	0.01 dB
4	Average crest factor of all segment packets	Real number	dB	0.01 dB
5	Average power upper limit test passed	Boolean	---	---
6	Average power lower limit test passed	Boolean	---	---
…	…			
	Results for last gate:			
1	Gate number	Integer	---	
2	Peak power recorded in segment	Real number	dBm	0.01 dB
3	Average power of all packets in segment	Real number	dBm	0.01 dB
4	Average crest factor of segment packets	Real number	dB	0.01 dB
5	Average power upper limit test passed	Boolean	---	---
6	Average power lower limit test passed	Boolean	---	---

Parameters

#	Description	Format	Units	Min	Max	Def
1	Segment index	Integer	---	---	---	---

Example of Use

```
:FETC:SRW:SUMM:WLAN:POW? 1
> 1,0,1,1,-9.59,-21.24,10.91,1,1
```

5

Remote Operation

:FETCh:SRWireless:SUMMary:WLAN:POWer? (Extended)

Function

Fetch WLAN transmit power (numeric) measurements – using extension code.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:POWer?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code	Integer	---	---
4	Number of gates	Integer	---	---
	Results for first gate:			
1	Gate number	Integer	---	---
2	Peak power recorded in segment	Real number	dBm	0.01 dB
3	Average power of all packets in segment	Real number	dBm	0.01 dB
4	Average crest factor of all segment packets	Real number	dB	0.01 dB
5	Average power upper limit test passed	Boolean	---	---
6	Average power lower limit test passed	Boolean	---	---
	If bit 0 of extension code set:			
7	Average power of packet with highest average power	Real number	dBm	0.01 dB
8	Average power of packet with lowest average power	Real number	dBm	0.01 dB
9	Packet index of packet with highest average power	Integer		
10	Packet index of packet with lowest average power			
…	…			
	Results for last gate:			
1	Gate number	Integer	---	
2	Peak power recorded in segment	Real number	dBm	0.01 dB
3	Average power of all packets in segment	Real number	dBm	0.01 dB
4	Average crest factor of segment packets	Real number	dB	0.01 dB
5	Average power upper limit test passed	Boolean	---	---

6	Average power lower limit test passed	Boolean	---	---
	If bit 0 of extension code set:			
7	Average power of packet with highest average power	Real number	dBm	0.01 dB
8	Average power of packet with lowest average power	Real number	dBm	0.01 dB
9	Packet index of packet with highest average power	Integer		
10	Packet index of packet with lowest average power			

Parameters

#	Description	Format	Units	Min	Max	Def
1	Segment index	Integer	---	---	---	---
2	[Extension code] (Optional)	Integer	---	0	$2^{32}-1$	0

Example of Use

```
:FETC:SRW:SUMM:WLAN:POW? 1, 1
> 1, 0, 1, 1, 1, -9.59, -21.24, 10.91, 1, 1, -18.32, -23.24, 7, 3
```

5

Remote Operation

:FETCh:SRWireless:SUMMary:WLAN:PPRofile:AVERage:ASCii?

Function

Fetch WLAN power profile (graphical) measurements - average - ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:PPRofile:AVERage:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Start time	Real number	s	1 µs
4	Stop time	Real number	s	1 µs
5	Number of points in power profile	Integer	---	---
6	First power value (averaged over segment)	Real number	dBm	0.01 dB
	…			
	Last power value (averaged over segment)	Real number	dBm	0.01 dB

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:PPR:AVER:ASC? 1
```

:FETCh:SRWireless:SUMMary:WLAN:PPRofile:AVERage[:BINary]?

Function

Fetch WLAN power profile (graphical) measurements - binary format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:PPRofile:AVERage[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary Header	---
2	Segment index	INT32	---
3	Packet index (always zero)	INT32	---
4	Start time	REAL32	seconds
5	Stop time	REAL32	seconds
6	Number of points	INT32	---
7	First power value (averaged over segment)	REAL32	dBm
	…		
	Last power value (averaged over segment)	REAL32	dBm

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:PPR:AVER? 1

> #48192<binary data - 8192 bytes long>
```

Notes:

1. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
2. Binary data will always be a multiple of 4 bytes.
3. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

5

Remote Operation

:FETCh:SRWireless:SUMMary:WLAN:PPRofile:PEAK:ASCii?

Function

Fetch WLAN power profile (graphical) measurements - peak - ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:PPRofile:PEAK:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Start time	Real number	s	1 µs
4	Stop time	Real number	s	1 µs
5	Number of points in power profile	Integer	---	---
6	First power value (averaged over segment)	Real number	dBm	0.01 dB
	…			
	Last power value (averaged over segment)	Real number	dBm	0.01 dB

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:PPR:PEAK:ASC? 1
```

:FETCh:SRWireless:SUMMary:WLAN:PPRofile:PEAK[:BINary]?

Function

Fetch WLAN power profile (graphical) measurements - peak - binary format.

Command

N/A

Query

`:FETCh:SRWireless:SUMMary:WLAN:PPRofile:PEAK[:BINary]?`

Response

#	Description	Format	Units
1	#nmm…mm	Binary Header	---
2	Segment index	INT32	---
3	Packet index (always zero)	INT32	---
4	Start time	REAL32	seconds
5	Stop time	REAL32	seconds
6	Number of points	INT32	---
7	First power value (averaged over segment)	REAL32	dBm
	…		
n	Last power value (averaged over segment)	REAL32	dBm

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

`:FETC:SRW:SUMM:WLAN:PPR:PEAK? 1`

Notes:

1. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
2. Binary data will always be a multiple of 4 bytes.
3. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

:FETCh:SRWireless:SUMMary:WLAN:PRAMps?

Function

Fetch WLAN power profile (numeric) measurements - power up and down ramps – WLAN packets

Note:

This command supersedes `:FETCh:SRWireless:SUMMary:WLAN:DSSS:PRAMps` following a change to support power ramp measurements on OFDM as well as DSSS signals. The use of the older command is now deprecated.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:PRAMps?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Average power-on ramp time for segment	Real number	s	0.01 µs
4	Average power-down ramp time for segment	Real number	s	0.01 µs
5	Power-on ramp time test passed	Boolean	---	---
6	Power-down ramp time test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:PRAM? 1
```

:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:GRAPh:ASCii?

Function

Fetch WLAN transmitter spectrum (graphical) measurements - ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:GRAPh:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Number of gates	Integer	---	---
	For first gate:			
1	Gate number	Integer	---	---
2	Bandwidth	Real number	Hz	1 Hz
3	Number of points in spectrum	Integer	---	---
4	First power value (averaged over segment)	Real number	dBm	0.01 dB
	…			
	Last power value (averaged over segment)	Real number	dBm	0.01 dB
	…			
	For last gate:			
1	Gate number	Integer	---	---
2	Bandwidth	Real number	Hz	1 Hz
3	Number of points in spectrum	Integer	---	---
4	First power value (averaged over segment)	Real number	dBm	0.01 dB
	…			
	Last power value (averaged over segment)	Real number	dBm	0.01 dB

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:SPEC:GRAP:ASC? 1
```

:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:GRAPh[:BINary]?

Function

Fetch WLAN transmitter spectrum (graphical) measurements - binary format.

Command

N/A

Query

`:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:GRAPh[:BINary]?`

Response

#	Description	Format	Units
1	#nmm…mm	Binary Header	---
2	Segment index	INT32	---
3	Packet index (always zero)	INT32	---
4	Number of gates	INT32	---
	For first gate:		
1	Gate number	INT32	---
2	Bandwidth	REAL32	Hz
3	Number of points	INT32	---
4	First power value	REAL32	dBm
…	…		
	Last power value	REAL32	dBm
…	…		
	For last gate:		
1	Gate number	INT32	---
2	Bandwidth	REAL32	Hz
3	Number of points	INT32	---
4	First power value	REAL32	dBm
…	…		
	Last power value	REAL32	dBm

Note:

The bandwidth fields can take the value (10, 70, 130 or 160 MHz, depending on the wireless standard. The x-axis for the graph data therefore runs from –BW/2 to +BW/2.

Parameters

#	Description	Format
1	Segment index	Number

Note:

1. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
2. Binary data will always be a multiple of 4 bytes.
3. Refer to sections 5.3.3 and 5.3.4 *Binary_Header_Information* and *Binary_Data_Fields*.

Example of Use

```
:FETC:SRW:SUMM:WLAN:SPEC:GRAP? 1
```

:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:GRAPh:COMBined:ASCii?

Function

Fetch WLAN transmitter spectrum (graphical) combined 802.11ac 80+80 measurements - ASCII format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:GRAPh:COMBined:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Number of gates	Integer	---	---
	For each gate:			
5	Gate number	Integer	---	---
6	Number of spectral profiles (See note)	Integer	---	---
	For each spectral profile			
7	Spectral profile number (1 or 2)	Integer		
8	Carrier frequency / carrier offset (See note)	Integer	Hz	1 Hz
9	Bandwidth	Integer	Hz	1 Hz
10	Number of points	Integer	---	---
	For each point			
11	Power value (averaged over segment)	Real number	dBm	0.01 dB

Notes:

The **number of spectral profiles** can be 1 or 2. If it is 1 it means that the two carriers forming the 80+80 transmission are less than or equal to 280 MHz apart. If it is 2 it means that the carriers are more than 280 MHz apart.

The **carrier frequency / carrier offset** field contains different information depending on the separation of the carriers.
If the carriers are less than or equal to 280 MHz apart, this value is the offset in Hz between the carrier frequency and the center of the profile. The center of the profile is defined as 0 Hz.
If the carriers are more than 280 MHz apart, this value is the frequency of the carrier.

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code (currently zero)	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:SPEC:GRAP:COMB:ASC? 1,2
```

:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:GRAPh:COMBined[:BINary]?

Function

Fetch WLAN transmitter spectrum (graphical) combined 802.11ac 80+80 measurements - binary format.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:GRAPh:COMBined[:BINary]?
```

Response

#	Description	Format	Units
1	#nmm…mm	Binary Header	---
2	Segment index	INT32	---
3	Packet index (always zero)	INT32	---
4	Extension code (currently zero)	INT32	---
5	Number of gates	INT32	---
	For each gate:		
6	Gate number	INT32	---
7	Number of spectral profiles (See notes)	INT32	---
	For each spectral profile	INT32	---
8	Spectral profile number (1 or 2)	INT32	---
9	Carrier frequency / carrier offset (See notes)	INT32	Hz
10	Bandwidth	INT32	Hz
11	Number of points	INT32	---
	For each point		
12	Power value (averaged over segment)	REAL32	dBm

Notes:

The **number of spectral profiles** can be 1 or 2. If it is 1 it means that the two carriers forming the 80+80 transmission are less than or equal to 280 MHz apart. If it is 2 it means that the carriers are more than 280 MHz apart.

The **carrier frequency / carrier offset** field contains different information depending on the separation of the carriers.
If the carriers are less than or equal to 280 MHz apart, this value is the offset in Hz between the carrier frequency and the center of the profile. The center of the profile is defined as 0 Hz.

If the carriers are more than 280 MHz apart, this value is the frequency of the carrier.

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code (currently zero)	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:SPEC:GRAP:COMB? 1,7
```

Notes:

1. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
2. Binary data will always be a multiple of 4 bytes.
3. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:MASK:COMBined:ASCii?

Function

Fetch WLAN transmitter spectrum mask for 802.11ac 80+80 measurements - ASCII format. The shape of the mask depends on the separation of the 80+80 carriers and the relative power level of the two signals. In the case of a summary measurement, the mask is calculated from spectral profiles of the individual packets captured in the segment.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:MASK:COMBined:ASCii?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code (currently zero)	Integer	---	---
4	Gate number	Integer	---	---
5	Number of spectral profiles (See note)	Integer	---	---
	For each spectral profile			
6	Spectral profile number (1 or 2)	Integer	---	---
7	Carrier frequency / carrier offset (See note)	Integer	Hz	1 Hz
8	Bandwidth	Integer	Hz	1 Hz
9	Number of points	Integer	---	---
	For each point			
10	Mask power value (averaged over segment)	Real number	dBm	0.01 dB

Notes:

The **number of spectral profiles** can be 1 or 2. If it is 1 it means that the two carriers forming the 80+80 transmission are less than or equal to 280 MHz apart. If it is 2 it means that the carriers are more than 280 MHz apart.

The **carrier frequency / carrier offset** field contains different information depending on the separation of the carriers.
If the carriers are less than or equal to 280 MHz apart, this value is the offset in Hz between the carrier frequency and the center of the profile. The center of the profile is defined as 0 Hz.
If the carriers are more than 280 MHz apart, this value is the frequency of the carrier.

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code (currently zero)	Number
3	Gate number	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:SPEC:MASK:COMB:ASC? 1,0,1
```

:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:MASK:COMBined[:BINary]?

Function

Fetch WLAN transmitter spectrum mask for 802.11ac 80+80 measurements - binary format. The shape of the mask depends on the separation of the 80+80 carriers and the relative power level of the two signals. In the case of a summary measurement, the mask is calculated from spectral profiles of the individual packets captured in the segment.

Command

N/A

Query

`:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:MASK:COMBined[:BINary]?`

Response

#	Description	Format	Units
1	#nmm...mm	Binary Header	---
2	Segment index	INT32	---
3	Packet index (always zero)	INT32	---
4	Extension code (currently zero)	INT32	---
5	Gate number	INT32	---
6	Number of spectral profiles (See notes)	INT32	---
	For each spectral profile	INT32	---
7	Spectral profile number (1 or 2)	INT32	---
8	Carrier frequency / carrier offset (See notes)	INT32	Hz
9	Bandwidth	INT32	Hz
10	Number of points	INT32	---
	For each point		
11	Power value (averaged over segment)	REAL32	dBm

Notes:

The **number of spectral profiles** can be 1 or 2. If it is 1 it means that the two carriers forming the 80+80 transmission are less than or equal to 280 MHz apart. If it is 2 it means that the carriers are more than 280 MHz apart.

The **carrier frequency / carrier offset** field contains different information depending on the separation of the carriers.

If the carriers are less than or equal to 280 MHz apart, this value is the offset in Hz between the carrier frequency and the center of the profile. The center of the profile is defined as 0 Hz.

If the carriers are more than 280 MHz apart, this value is the frequency of the carrier.

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code (currently zero)	Number
3	Gate number	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:SPEC:MASK:COMB? 1,0,1
```

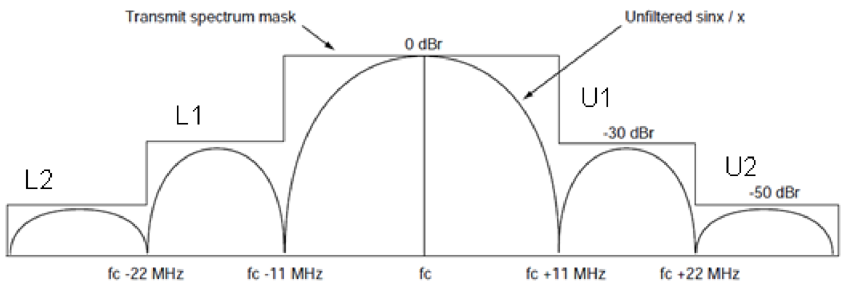
Notes:

1. Always analyze the header to determine the length of the binary data. (Do not rely on the length remaining unchanged following a software update.)
2. Binary data will always be a multiple of 4 bytes.
3. Refer to sections 5.3.3 and 5.3.4 *Binary Header Information* and *Binary Data Fields*.

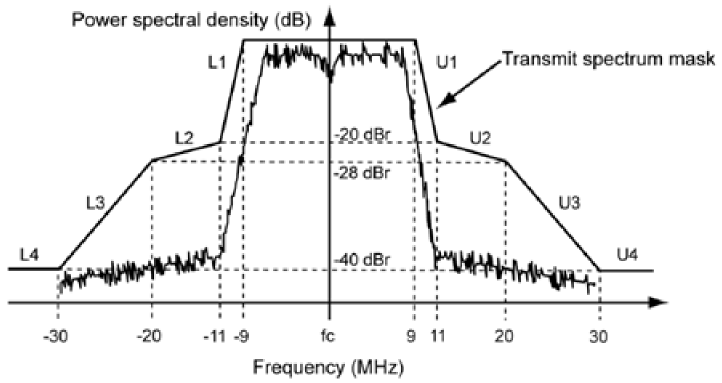
:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:NUMeric?

Function

Fetch WLAN transmitter spectrum (numeric) measurements.
Limit masks are divided into segments. Segments below the carrier have the designation ‘L’ for ‘Lower’; and those above the carrier are designated ‘U’ for ‘Upper’. DSSS masks have two segments above and below the carrier: L2, L1, U1 and U2.



OFDM masks have four segments above and below the carrier: L4 to L1 and U1 to U4.



Measurements are returned for each segment of the limit mask as follows:

- The frequency offset in Hz from the carrier to the highest signal peak within the mask segment.
- The power of the signal peak relative to the limit mask at that frequency offset.
- The absolute power of the signal peak at that frequency offset.

The summary measurement returns the worst case results for each mask segment and lists the packet indices corresponding to the worst case results.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:NUMeric?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Number of gates	Integer	---	---
	Results for each gate:			
1	Gate number	Integer	---	---
2	RF carrier suppression (DSSS) - average for segment	Real number	dB	0.01 dB
3	Power spectral density - average for segment	Real number	dBm / MHz	0.01 dB
4	Occupied bandwidth - average for segment	Real number	Hz	1 Hz
5	Spectrum mask passed limit test	Boolean	---	---
6	RF carrier suppression (DSSS) test passed	Boolean	---	---
7	Power spectral density test passed	Boolean	---	---
8	Occupied bandwidth test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:SPEC:NUM 1
```

:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:NUMeric? (Extended)

Function

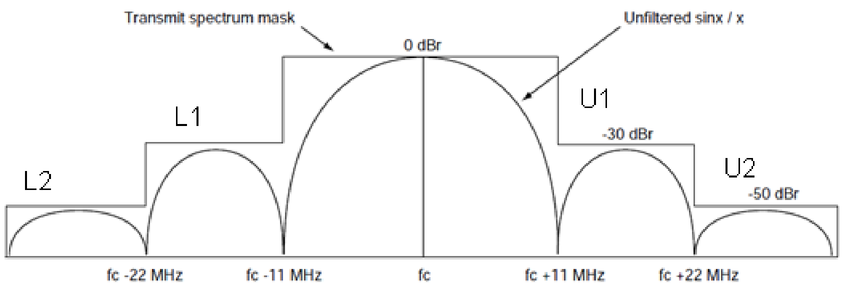
Fetch WLAN transmitter spectrum (numeric) measurements with optional extension code.

Extension code bit	Additional fields in response
0	Worst case mask margin results considering each packet within the capture individually.
1	Mask violation percentage for the worst case packet.
2	Power spectral density (averaged over segment) in units of mW / MHz. Sum of power spectral density results. (Sequential MIMO only)
3	Mask margin and mask violation results based on the average spectral response of all packets in the capture.
4	Extends the mask margin results (selected by bit 0 and bit 3) to include results for the additional mask segment applicable to 802.11p regulatory masks associated with on of the four power classes (A to D). This bit should only be set when measuring an 802.11p signal against the specification defined by the power class.

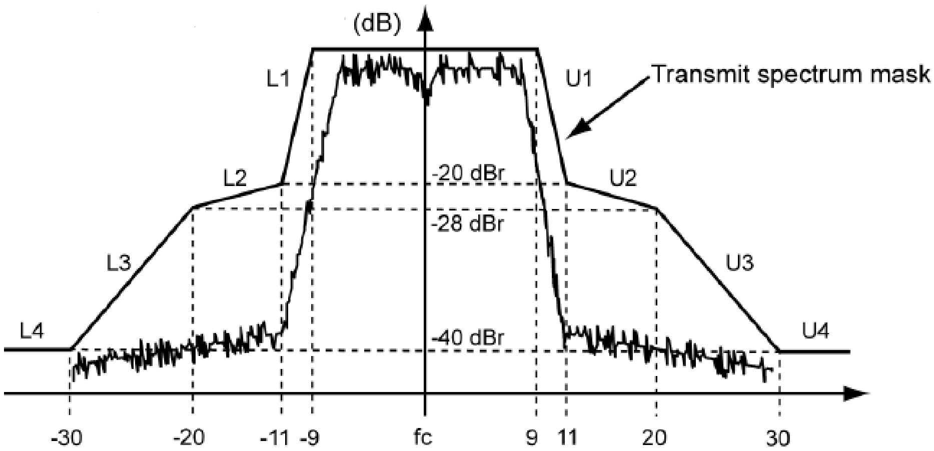
These measurement options are described in detail below:

Mask Margin Results

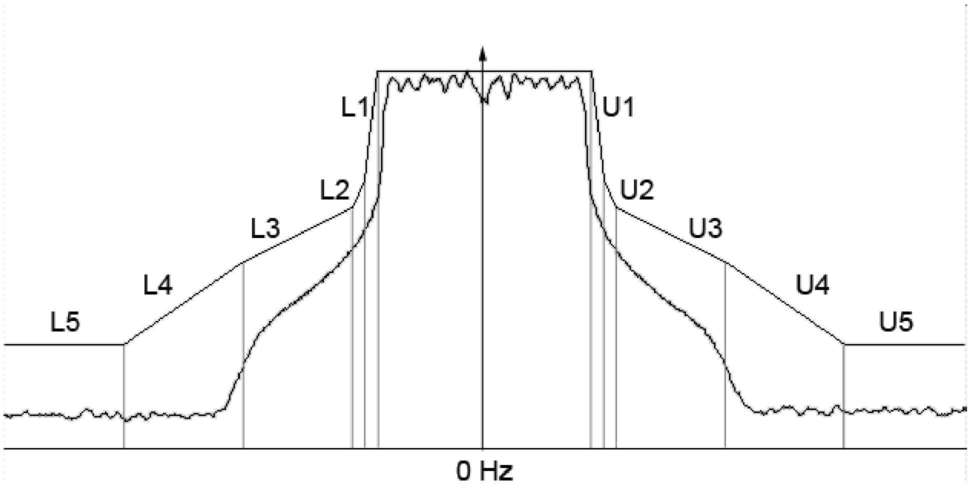
Limit masks are divided into segments. Segments below the carrier have the designation 'L' for 'Lower'; and those above the carrier are designated 'U' for 'Upper'. DSSS masks defined in the IEEE 802.11 standard have two segments above and below the carrier: L2, L1, U1 and U2.



OFDM masks defined in the IEEE 802.11 standard have four segments above and below the carrier: L4 to L1 and U1 to U4.



802.11p regulatory masks associated with one of the four power classes: A, B, C or D have an additional pair of segments as shown in the figure and table below.



	Frequency offset from carrier (MHz)									
	L5	**L4**	**L3**	**L2**	**L1**	**U1**	**U2**	**U3**	**U4**	**U5**
5 MHz	–7.5	–5	–2.75	–2.5	–2.25	+2.25	+2.5	+2.75	+5	+7.5
10 MHz	–15	–10	–5.5	–5	–4.5	+4.5	+5	+5.5	+10	+15
20 MHz	–30	–20	–11	–10	–9	+9	+10	+11	+20	+30

The table above shows the frequency offsets from the carrier of the start of each segment of the 802.11p spectral mask for each signal bandwidth. Note that there are 10 segments (L5 to U5) compared with the IEEE standard mask, which has 8 segments. Remember to set bit 4 of the extension code when measuring 802.11p signals against the regulatory masks. This will ensure that results for all 10 segments appear in the response.

Mask Margin measurements are returned for each segment of the limit mask as follows:

- The frequency offset in Hz from the carrier to the signal peak within the mask segment that is closest to the mask or exceeds the mask by the greatest amount.
- The power of the signal peak relative to the limit mask at that frequency offset.
- The absolute power of the signal peak at that frequency offset.

If bit 0 of the extension code is set, the summary measurement returns the worst case mask margin results for each mask segment and lists the packet indices corresponding to the worst case results.

Note that these results may indicate mask violations on individual packets due to noise, while the overall spectral mask test, based on the average of all packets in a capture, indicates a pass.

If bit 3 of the extension code is set, the summary measurement returns the mask margin result calculated against the average of the spectral responses of all packets in the capture.

Mask Violation Percentage (Worst Case Packet)
If bit 1 of the extension code is set, the instrument returns the worst case mask violation and the packet number corresponding to the worst case. Mask violation is the proportion of the trace that exceeds the mask limit. It is used to indicate the “degree of failure”. If the mask is crossed at only a few frequency points, the violation percentage value is small. A more serious breach of the limit will result in a higher violation percentage value.

Power Spectral Density (mW / MHz)
If bit 2 of the extension code is set the instrument returns the power spectral density result in units of mW / MHz as well as the equivalent result in dBm / MHz, which is always returned in the response. A second field returns the sum of the spectral power density values (mW / MHz) for all sequential MIMO segments in the capture. (If the capture is not sequential MIMO, the value returned is identical to the preceding mW / MHz value.)

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:NUMeric?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code	Integer	---	---
4	Number of gates	Integer	---	---
	Results for each gate:			
1	Gate number	Integer	---	---
2	RF carrier suppression (DSSS)	Real number	dB	0.01 dB
3	Power spectral density	Real number	dBm / MHz	0.01 dB
	If bit 2 of extension code set:			
4	Power spectral density (mW)	Real number	mW / MHz	3 sig figs
5	Sum of power spectral density results for all sequential MIMO segments. See notes	Real number	mW / MHz	3 sig figs
6	Occupied bandwidth	Real number	Hz	1 Hz

	If bit 3 of extension code set:			
	Mask margin results from summary average			
	If bit 4 of extension code is set (as well as bit 3)			
7	L5 freq offset from carrier of summary avg max	Real number	Hz	1 Hz
8	L5 power relative to mask of summary avg max	Real number	dB	0.01 dB
9	L5 absolute power of summary avg max	Real number	dBm	0.01 dB
10	L4 freq offset from carrier of summary avg max	Real number	Hz	1 Hz
11	L4 power relative to mask of summary avg max	Real number	dB	0.01 dB
12	L4 absolute power of summary avg max	Real number	dBm	0.01 dB
13	L3 freq offset from carrier of summary avg max	Real number	Hz	1 Hz
14	L3 power relative to mask of summary avg max	Real number	dB	0.01 dB
15	L3 absolute power of summary avg max	Real number	dBm	0.01 dB
16	L2 freq offset from carrier of summary avg max	Real number	Hz	1 Hz
17	L2 power relative to mask of summary avg max	Real number	dB	0.01 dB
18	L2 absolute power of summary avg max	Real number	dBm	0.01 dB
19	L1 freq offset from carrier of summary avg max	Real number	Hz	1 Hz
20	L1 power relative to mask of summary avg max	Real number	dB	0.01 dB
21	L1 absolute power of summary avg max	Real number	dBm	0.01 dB
22	U1 freq offset from carrier of summary avg max	Real number	Hz	1 Hz
23	U1 power relative to mask of summary avg max	Real number	dB	0.01 dB
24	U1 absolute power of summary avg max	Real number	dBm	0.01 dB
25	U2 freq offset from carrier of summary avg max	Real number	Hz	1 Hz
26	U2 power relative to mask of summary avg max	Real number	dB	0.01 dB
27	U2 absolute power of summary avg max	Real number	dBm	0.01 dB
28	U3 freq offset from carrier of summary avg max	Real number	Hz	1 Hz
29	U3 power relative to mask of summary avg max	Real number	dB	0.01 dB
30	U3 absolute power of summary avg max	Real number	dBm	0.01 dB
31	U4 worst case - frequency offset from carrier	Real number	Hz	1 Hz
32	U4 worst case - power relative to mask	Real number	dB	0.01 dB
33	U4 absolute power of summary avg max	Real number	dBm	0.01 dB
	If bit 4 of extension code is set (as well as bit 3)			
34	U5 worst case - frequency offset from carrier	Real number	Hz	1 Hz
35	U5 worst case - power relative to mask	Real number	dB	0.01 dB
36	U5 absolute power of summary avg max	Real number	dBm	0.01 dB
37	Mask violation of summary average	Real number	%	0.01%

	If bit 0 of extension code set: Mask margin results from worst case packets in the capture			
	If bit 4 of extension code is set (as well as bit 0)			
38	L5 worst case - frequency offset from carrier	Real number	Hz	1 Hz
39	L5 worst case - power relative to mask	Real number	dB	0.01 dB
40	L5 worst case - absolute power	Real number	dBm	0.01 dB
41	L4 worst case - frequency offset from carrier	Real number	Hz	1 Hz
42	L4 worst case - power relative to mask	Real number	dB	0.01 dB
43	L4 worst case - absolute power	Real number	dBm	0.01 dB
44	L3 worst case - frequency offset from carrier	Real number	Hz	1 Hz
45	L3 worst case - power relative to mask	Real number	dB	0.01 dB
46	L3 worst case - absolute power	Real number	dBm	0.01 dB
47	L2 worst case - frequency offset from carrier	Real number	Hz	1 Hz
48	L2 worst case - power relative to mask	Real number	dB	0.01 dB
49	L2 worst case - absolute power	Real number	dBm	0.01 dB
50	L1 worst case - frequency offset from carrier	Real number	Hz	1 Hz
51	L1 worst case - power relative to mask	Real number	dB	0.01 dB
52	L1 worst case - absolute power	Real number	dBm	0.01 dB
53	U1 worst case - frequency offset from carrier	Real number	Hz	1 Hz
54	U1 worst case - power relative to mask	Real number	dB	0.01 dB
55	U1 worst case - absolute power	Real number	dBm	0.01 dB
56	U2 worst case - frequency offset from carrier	Real number	Hz	1 Hz
57	U2 worst case - power relative to mask	Real number	dB	0.01 dB
58	U2 worst case - absolute power	Real number	dBm	0.01 dB
59	U3 worst case - frequency offset from carrier	Real number	Hz	1 Hz
60	U3 worst case - power relative to mask	Real number	dB	0.01 dB
61	U3 worst case - absolute power	Real number	dBm	0.01 dB
62	U4 worst case - frequency offset from carrier	Real number	Hz	1 Hz
63	U4 worst case - power relative to mask	Real number	dB	0.01 dB
64	U4 worst case - absolute power	Real number	dBm	0.01 dB
	If bit 4 of extension code is set (as well as bit 0)			
65	U4 worst case - frequency offset from carrier	Real number	Hz	1 Hz
66	U4 worst case - power relative to mask	Real number	dB	0.01 dB
67	U4 worst case - absolute power	Real number	dBm	0.01 dB
68	L5 worst case packet number	Integer	---	---

69	L4 worst case - packet number	Integer	---	---
70	L3 worst case - packet number	Integer	---	---
71	L2 worst case - packet number	Integer	---	---
72	L1 worst case - packet number	Integer	---	---
73	U1 worst case - packet number	Integer	---	---
74	U2 worst case - packet number	Integer	---	---
75	U3 worst case - packet number	Integer	---	---
76	U4 worst case - packet number	Integer	---	---
	If bit 4 of extension code is set (as well as bit 0)			
77	U5 worst case - packet number	Integer	---	---

	If bit 1 of extension code set:			
78	Worst case mask violation	Real number	%	0.01 %
79	Worst case mask violation packet number	Integer	---	---

	The following fields are always present:			
80	Spectrum mask passed limit test	Boolean	---	---
81	RF carrier suppression (DSSS) passed	Boolean	---	---
82	Power spectral density test passed	Boolean	---	---
83	Occupied bandwidth test passed	Boolean	---	---

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:SPEC:NUM? 1,1
```

:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:SPOT?

Function

Fetch spot spectrum measurements at a frequency offset above and below the carrier frequency (f_c).

Note that the specified frequency must be one of those previously defined using :CONFigure:SRWireless:WLAN:SPOT:FREQuency:ADD.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:SPECtrum:SPOT?
```

Response

#	Description	Format	Units	Resolution
1	Segment index	Integer	---	---
2	Packet index (always zero)	Integer	---	---
3	Extension code (reserved – always zero)	Integer	---	---
4	Number of gates	Integer	---	---
	Results for first gate:			
1	Gate number	Integer	---	---
2	Frequency offset	Real number	Hz	1 Hz
3	Absolute power at f_c + offset (averaged over segment)	Real number	dBm	0.01 dB
4	Relative power compared to limit mask at f_c + offset (averaged over segment)	Real number	dB	0.01 dB
5	Relative power compared to signal peak at f_c + offset (averaged over segment)	Real number	dB	0.01 dB
6	Peak power at f_c + offset (across all packets in segment)	Real number	dBm	0.01 dB
7	Worst case relative power compared to limit mask at f_c + offset (across all packets in segment)	Real number	dB	0.01 dB
8	Absolute power at f_c - offset (averaged over segment)	Real number	dBm	0.01 dB
9	Relative power compared to limit mask at f_c – offset (averaged over segment)	Real number	dB	0.01 dB
10	Relative power compared to signal peak at f_c – offset (averaged over segment)	Real number	dB	0.01 dB
11	Peak power at f_c – offset (across all packets in segment)	Real number	dBm	0.01 dB
12	Worst case relative power compared to limit mask at f_c – offset (across all packets in segment)	Real number	dB	0.01 dB
13	Packet index having worst case relative power compared to limit mask at f_c + offset.	Integer	---	---

14	Packet index having worst case relative power compared to limit mask at f_c – offset	Integer	---	---
…	…			
	Results for last gate:			
1	Gate number	Integer	---	---
2	Frequency offset	Real number	Hz	1 Hz
3	Absolute power at f_c + offset (averaged over segment)	Real number	dBm	0.01 dB
4	Relative power compared to limit mask at f_c + offset (averaged over segment)	Real number	dB	0.01 dB
5	Relative power compared to signal peak at f_c + offset (averaged over segment)	Real number	dB	0.01 dB
6	Peak power at f_c + offset (across all packets in segment)	Real number	dBm	0.01 dB
7	Worst case relative power compared to limit mask at f_c + offset (across all packets in segment)	Real number	dB	0.01 dB
8	Absolute power at f_c - offset (averaged over segment)	Real number	dBm	0.01 dB
9	Relative power compared to limit mask at f_c – offset (averaged over segment)	Real number	dB	0.01 dB
10	Relative power compared to signal peak at f_c + offset (averaged over segment)	Real number	dB	0.01 dB
11	Peak power at f_c – offset (across all packets in segment)	Real number	dBm	0.01 dB
12	Worst case relative power compared to limit mask at f_c – offset (across all packets in segment)	Real number	dB	0.01 dB
13	Packet index having worst case relative power compared to limit mask at f_c + offset.	Integer	---	---
14	Packet index having worst case relative power compared to limit mask at f_c – offset	Integer	---	---

Parameters

#	Description	Format	Units	Min	Max	Def	Resolution
1	Segment index	Number	---	---	---	---	---
2	Extension code	Number	---	0	2^{32}–1	0	---
3	Frequency offset	Number	Hz	0	See note	0	1 Hz

Note:

The maximum frequency offset is equal to half the spectrum bandwidth. The spectrum bandwidth depends on the wireless standard.

Example of Use

```
:FETC:SRW:SUMM:WLAN:SPEC:SPOT? 1,0,20E+06  // Extract spot readings
                                           // at ±20 MHz.
> 1,0,0,1,1,20000000,
  -49.95,-12.03,-39.76,-47.00,-10.16       // fc + 20 MHz
  -49.87,-12.12,-39.65,-47.16,-10.08       // fc – 20 MHz
  7, 7                                     // Worst case packets
```

:FETCh:SRWireless:SUMMary:WLAN:SPOWer?

Function

Fetch the composite MIMO stream power and related measurements.

Command

N/A

Query

```
:FETCh:SRWireless:SUMMary:WLAN:SPOWer?
```

Response

#	Description	Format	Units
1	Segment index	Integer	---
2	Packet index (always zero)	Integer	---
3	Extension code	Integer	---
4	Number of spatial streams	Integer	---
	For first spatial stream:		
1	Average stream power for all packets in segment	Real number	dBm
2	Peak power recorded in segment	Real number	dBm
3	Average stream crest factor for all packets in segment	Real number	dB
…			
	For last spatial stream:		
1	Average stream power for all packets in segment	Real number	dBm
2	Peak power recorded in segment	Real number	dBm
3	Average stream crest factor for all packets in segment	Real number	dB

Parameters

#	Description	Format
1	Segment index	Number
2	Extension code	Number

Example of Use

```
:FETC:SRW:SUMM:WLAN:SPOW? 1,0
```

:INITiate:SRWireless

Function

Initiate capture and analysis.

An optional Capture Options parameter may be used to control how the capture and analysis functions are performed

Bit 0	Perform automatic leveling before the capture.
Bits 1 to 31	Reserved for future use.

Command

```
:INITiate:SRWireless
```

Query

N/A

Response

N/A

Parameters

#	Description	Format	Units	Min	Max	Def	Resolution
1	Capture options (optional)	Integer	---	0	2^32–1	0	---

Example of Use

```
:INIT:SRW              // Initiate a capture without automatic leveling.
*WAI


:CONF:SRW:ALEV:TIME 0.002    // Set the capture time for automatic leveling.
:INIT:SRW 1            // Initiate capture with automatic leveling.
*WAI
```

:INITiate:SRWireless:ALEVel

Function

Initiate the automatic leveling function.

Command

```
:INITiate:SRWireless:ALEVel
```

Query

N/A

Response

N/A

Parameters

N/A

Example of Use

```
:INIT:SRW:ALEV                     // Start automatic leveling function.
*WAI                               // Wait for completion.
:CONF:SRW:POW?                     // Read back the level that was
                                   // determined by the leveling function
> -12.34
```

:STATus:SRWireless:MEASurement?

Function

Check the status of a measurement in progress.

Command

N/A

Query

```
:STATus:SRWireless:MEASurement?
```

Response

#	Description	Format
1	Measurement status	Integer

Note:

The value returned by this query is an unsigned 32-bit integer. The following table describes the meaning of each bit.

Bit	Name	Description
0	Measurement complete	This bit is cleared when a measurement is started and set when the measurement system has terminated. The Measurement terminates when measurements are complete, aborted or stopped due to an incorrect configuration or trigger timeout.
1	Trigger ready	This bit is set when the capturing system reports that it is ready to start capturing and cleared when the capture begins.
2	Trigger timeout	This bit is set if a trigger does not occur within the trigger timeout period (see :CONFigure:SRWireless:TTIMeout.), and indicates that no measurements were made.
3 - 32	Reserved	Reserved for future use.

Parameters

N/A

Example of Use

```
:STAT:SRW:MEAS?        // Check measurement status
> 1                    // Bit zero is set:  Measurement complete.
```

:STATus:SRWireless:TMIMo:CONNect?

Function

Queries the status of all connections to MU887000A modules used for a True MIMO measurement.
Currently no extension codes have been defined for this command. Use zero.

Command

N/A

Query

```
:STATus:SRWireless:TMIMo:CONNect?
```

Response

#	Description	Format
1	Extension code	Integer
2	Overall connection status flag	Boolean
3	Number of TMIMOS segments	Integer
	For each TMIMOS segment	
1	Segment number	Integer
2	IP address of corresponding “S” module	String
3	Connection status flag	Boolean
4	Error code reported by “S” module	Integer

Error codes:

0	No error
1	Command Error
2	Execution Error
3	Timeout

Parameters

#	Description	Format	Units	Min	Max	Default
1	Extension code	Number	---	---	---	0

Example of Use

```
:STAT:SRW:TMIM:CONN? 0

> 0,1,1,2,”192.168.50.3”,1,0
```

Interpretation:

`0`	Extension code
`1`	Overall connection status (1 = True, implies that all required connections have been established.)
`1`	One TMIMOS segment is defined
`2`	Segment number is 2
`"192.168.50.3"`	IP address of "S" module corresponding to the TMIMOS segment.
`1`	Connection status of this segment. (1 = True – Connected)
`0`	No errors reported by "S" module.

Chapter 6 Measurement Examples

This chapter provides examples to show how the remote SCPI commands can be combined to perform a number of common operations.

6.1 Selecting the SRW Application

Command	Description
`SYST:LANG SCPI`	Ensure that the instrument is set to receive SCPI commands.
`INST:SEL SRW`	Select the short range wireless application.

6.2 Routing RF Connections

Command	Description
`ROUT:PORT:CONN:DIR <input>, <output>`	Select the front panel connectors to be used for the measurement. <input> and <output> can be: PORT1 PORT2 PORT3 PORT4

6.3 Setting up the Trigger

The operation of the trigger is described in section 2.4.

When a capture segment is created, the default trigger mode is IMMEDIATE. This is the simplest trigger mode to use in cases where the DUT is producing a continuous stream of packets of the same type.

You should use LEVEL trigger mode if:

- the DUT has been configured to generate a fixed number of packets; or
- the DUT has been configured to generate a list of different packet types; or
- you wish to be certain to capture and measure the first packet that the DUT produces.

When using LEVEL trigger mode, consider setting a trigger delay to ensure that the system captures the rising edge of the packet. You should also consider setting a trigger timeout.

Command	Description
CONF:SRW:TRIG IMMEDIATE	Immediate Triggering Set up immediate triggering. Capture will start when the instrument receives an `INIT:SRW` command. This is the default setting when a capture segment is created.
CONF:SRW:TRIG LEVEL	Level Triggering Set up level triggering.
CONF:SRW:TLEV -20	Set the trigger level in dB. The trigger level is set with respect to the expected input power level. In this example, the capture will start when the RF input level exceeds the expected input power level setting minus 20 dB. So if the expected input level is set to +13 dBm, the capture will trigger when the signal input exceeds +13 – 20 = –13 dBm. The trigger level is –20 dB by default, which works well for most WLAN and *Bluetooth* signals.
CONF:SRW:TDEL -1E-05	Set the trigger delay to –10 μs. Setting a negative trigger delay of a few microseconds ensures that the whole of the packet's power-on ramp is captured.
CONF:SRW:TTIM 1	Set the trigger timeout in seconds. This sets the time that the system will wait for a trigger after receiving the `INIT:SRW` command. You can find out whether a trigger timeout occurred by checking whether bit 2 of the response returned by `:STAT:SRW:MEAS?` is set. If the trigger timeout is set to zero, the system will wait indefinitely for a trigger.

6.4 Initiating Captures and Checking Results

Captures are initiated using the INIT:SRW command. There are a number of methods to determine when the capture is complete, as shown in the following table.

Command	**Description**

`INIT:SRW`
`loop`
`  short delay`
`  STAT:SRW:MEAS?`
`until bit 0 of response is '1'`
	Polling Measurement Status
	STAT:SRW:MEAS? returns a 32-bit unsigned integer, with bit-0 set when the measurement is complete.
	The bit is cleared at power-on and in response to *RST or an INIT:SRW command.
	Include a short delay (a few ms) in the loop to prevent the instrument from being polled continuously.

`INIT:SRW` `*WAI`	Using the IEEE-488.2 Common Command *WAI Commands that follow *WAI are held until the capture has completed. If you are controlling the instrument via a VISA session or other mechanism that implements timeouts, ensure that the timeout is set longer than the expected capture time.
`INIT:SRW` `*OPC?`	Using the IEEE-488.2 Common Command *OPC? When the capture is complete the instrument places a ‘1’ character in the output queue. If you are controlling the instrument via a VISA session or other mechanism that implements timeouts, ensure that the timeout is set longer than the expected capture time

`*CLS`
`INIT:SRW`
`*OPC`
`loop`
`  short delay`
`  *ESR?`
`until bit 0 of response is '1'`
	Using the IEEE-488.2 Common Command *OPC in conjunction with *ESR?
	When the capture is complete, the system sets the OPC event bit (bit-0) in the Standard Event Status Register. This may be polled using the query command *ESR?
	Include a short delay (a few ms) in the loop to prevent the instrument from being polled continuously.
	The *CLS command ensures that the event status register is cleared before the *OPC command is sent.

6.5 Aborting Captures

Script	Description

`ABOR:SRW`
`loop`
`  short delay`
`  STAT:SRW:MEAS?`
`until bit 0 of response is '1'`
	The ABORt:SRWireless command may be used to cancel a capture.
	The simplest way to use this command is in conjunction with STAT:SRW:MEAS? This returns an unsigned 32-bit integer with bit-0 set when measurement is complete or aborted.
	The bit is cleared at power-on and in response to *RST or an INIT:SRW command.

6

Measurement Examples

6.6 CW Power and Frequency Measurements

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td>CONF:SRW:SEGM:CLE</td><td>Clear any existing capture segments.</td></tr>
<tr><td>CONF:SRW:SEGM:APP CW</td><td>Create a CW capture segment.</td></tr>
<tr><td>CONF:SRW:TIME 1E-03</td><td>Set the capture time to 1 ms.</td></tr>
<tr><td>CONF:SRW:FREQ 2.412E+09</td><td>Set the frequency to 2.412 GHz</td></tr>
<tr><td>CONF:SRW:POW -10</td><td>Set the expected power level to –10 dBm</td></tr>
<tr><td>CONF:SRW:SEL:CW:POW ON
CONF:SRW:SEL:CW:FREQ ON</td><td>Enable CW power and frequency measurements.
All measurements are disabled by default when a new capture is created.</td></tr>
<tr><td>Refer to Section 6.4</td><td>Initiate the capture and wait for it to complete.</td></tr>
<tr><td>Refer to Section 7.1</td><td>Check that no errors occurred during the measurement.</td></tr>
<tr><td>FETC:SRW:SUMM:CW:POW? 1</td><td>Read the summary power measurement result
Example: 1,0,–11.12,–10.15
<table>
<tr><td>1</td><td>Segment index</td></tr>
<tr><td>0</td><td>Packet index (always zero for summary measurements.)</td></tr>
<tr><td>–11.12</td><td>Average power in dBm recorded over the capture time of 2 ms.</td></tr>
<tr><td>–10.15</td><td>Peak power in dBm recorded over the capture time of 2 ms.</td></tr>
<tr><td>1</td><td>Average power upper limit test passed.</td></tr>
<tr><td>1</td><td>Average power lower limit test passed.</td></tr>
</table></td></tr>
<tr><td>FETC:SRW:SUMM:CW:FREQ? 1,1</td><td>Read the summary frequency measurement result using the extension code to return both the offset between the VSA frequency and measured frequency and the absolute frequency. It is assumed that the VSA has been tuned to 2.412 GHz.

Example: 0,0,1,123456,2412123456,1
<table>
<tr><td>0</td><td>Segment number</td></tr>
<tr><td>0</td><td>Packet number (always zero for summary measurements.)</td></tr>
<tr><td>1</td><td>Extension code. Bit 0 is set.</td></tr>
<tr><td>123456</td><td>Frequency offset in Hz measured over the capture time of 2 ms.</td></tr>
<tr><td>2412123456</td><td>Absolute frequency in Hz measured over the capture time of 2 ms.</td></tr>
<tr><td>1</td><td>Frequency limit test passed.</td></tr>
</table></td></tr>
</table>

Note:

Note that the default limits for CW power and frequency measurements are set to values that guarantee a pass. To set the limits use:

CONF:SRW:CW:POW:ULIM <upper limit in dBm>
CONF:SRW:CW:POW:LLIM <lower limit in dBm>
CONF:SRW:CW:FREQ:LIM <frequency offset limit in Hz>

6.7 Continuous WLAN Measurements

6.7.1 Numeric Measurement Example – EVM

Command	**Description**
`CONF:SRW:SEGM:CLE`	Clear any existing capture segments.
`CONF:SRW:SEGM:APP WLG`	Create a capture segment for WLAN 802.11g packets.
`CONF:SRW:TIME 5E-03`	Set the capture time to 5 ms Set a capture time sufficiently long to capture at least the number of packets required for the measurement.
`CONF:SRW:FREQ 2.5E+09`	Set the frequency to 2.5 GHz.
`CONF:SRW:POW -10`	Set the expected power level.
`CONF:SRW:PACK 1`	Set the number of packets to be measured.
`CONF:SRW:WLG:DRAT 54MBPS`	Set the expected data rate of the packets.
`CONF:SRW:SEL:WLAN:EVM ON`	Enable EVM measurements
Refer to Section 6.3	Set up the trigger
Refer to Section 6.4	Initiate the capture and wait for it to complete.
Refer to Section 7.1	Check that no errors occurred during the measurement.
`FETC:SRW:SUMM:WLAN:OFDM:EVM? 1`	Read the EVM measurement summary results. Example:

1	Segment index
0	Packet index (always zero for summary measurements.)
64	Number of subcarriers
38	Number of symbols
–39.6003	Centre frequency leakage
–0.23	Symbol clock frequency tolerance (ppm)
0.124969	Centre frequency tolerance (ppm)
300.176	Centre frequency tolerance (Hz)
–38.2331	Max EVM in segment (dB)
1.22559	Max EVM in segment (%)
–62.579	Min EVM (dB)
0.0744916	Min EVM (%)
–45.7253	Average EVM (dB)
0.517289	Average EVM (%)
1	Centre frequency leakage test pass flag
1	Symbol clock frequency tolerance test pass flag
1	Centre frequency leakage test pass flag
1	EVM test pass flag
1	Spectral flatness test pass flag
1	Packet number for max EVM
1	Packet number for min EVM

6.7.2 Interpreting Spectral Flatness

After capturing a stream of 802.11g packets with EVM measurements enabled, it is possible to extract ‘advanced’ EVM measurements comprising numeric tables that may be plotted as graphs as required.

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td><code>FETC:SRW:SUMM:WLAN:OFDM:AEVM? 1</code></td><td>Extract the advanced EVM measurement summary results.
Example:
<table>
<tr><td>1</td><td>Segment index</td></tr>
<tr><td>0</td><td>Packet index (always zero for summary measurements.)</td></tr>
<tr><td></td><td>Number of symbols</td></tr>
<tr><td></td><td>Number of data subcarriers</td></tr>
<tr><td></td><td>Number of pilot subcarriers</td></tr>
<tr><td></td><td>Number of subcarriers</td></tr>
<tr><td></td><td>First element of spectral flatness array.</td></tr>
<tr><td>…</td><td>…</td></tr>
<tr><td></td><td>Last element of spectral flatness array.</td></tr>
<tr><td></td><td>First element of EVM per data subcarrier array.</td></tr>
<tr><td>…</td><td>…</td></tr>
<tr><td></td><td>Last element of EVM per data subcarrier array.</td></tr>
<tr><td></td><td>First element of EVM per pilot subcarrier array.</td></tr>
<tr><td>…</td><td>…</td></tr>
<tr><td></td><td>Last element of EVM per pilot subcarrier array.</td></tr>
<tr><td></td><td>First element of EVM per symbol array.</td></tr>
<tr><td>…</td><td>…</td></tr>
<tr><td></td><td>Last element of EVM per symbol array.</td></tr>
</table>
</td></tr>
</table>

Note:

To interpret the array data correctly, refer to the following fields in the response:

Number of symbols: length of the EVM per symbol array

Number of subcarriers: length of the spectral flatness array.

Number of data subcarriers: length of the EVM per data subcarrier array.

Number of pilot subcarriers: length of the EVM per pilot subcarrier array.

6.7.3 Interpreting the Spectral Flatness Data Array

The spectral flatness data returned by the 802.11g EVM (Numeric) measurement shows the power in dBm of each subcarrier in the OFDM signal.

An 802.11g signal has 64 subcarriers arranged about the carrier frequency. By convention the subcarriers at frequencies below the carrier are labeled –32 to –1, and those above are labeled +1 to +31. Subcarriers –26 to –1 and +1 to +26 are modulated; the others are not used.

Four of the modulated subcarriers are pilots: –21, –7, +7 and +21. The remaining modulated subcarriers carry data.

The results array has 64 elements. Element 0 corresponds to subcarrier –32, element 1 corresponds to subcarrier –31 and so on. Note that the array will contain zeros at the positions corresponding to the unused subcarriers (shown in grey in the table below). At other positions it will contain the power reading for the subcarrier. At the carrier frequency position, the power reading represents the carrier leakage.

The following table shows typical data for an 802.11g EVM measurement:

Index	Data	Sub-carrier	Index	Data	Sub-carrier
0	0	–32	32	–22.5372	0 CARRIER
1	0	–31	33	0.0623136	+1
2	0	–30	34	0.0368384	+2
3	0	–29	35	0.0729722	+3
4	0	–28	36	0.00133916	+4
5	0	–27	37	0.044085	+5
6	–0.975336	–26	38	0.00227012	+6
7	–0.619161	–25	39	0.00296693	+7 PILOT
8	–0.422396	–24	40	0.00617417	+8
9	–0.254359	–23	41	0.0120763	+9
10	–0.185222	–22	42	–0.00485009	+10
11	–0.0854975	–21 PILOT	43	–0.0127595	+11
12	–0.0716982	–20	44	0.00534614	+12
13	–0.0858615	–19	45	0.0291396	+13
14	–0.0724213	–18	46	–0.00217479	+14
15	–0.0900968	–17	47	–0.0519658	+15
16	–0.0544801	–16	48	–0.0453166	+16
17	–0.0626069	–15	49	–0.0291117	+17
18	–0.0320724	–14	50	–0.0432968	+18
19	–0.0269051	–13	51	–0.0475234	+19
20	–0.00352104	–12	52	–0.0690436	+20
21	–0.0424398	–11	53	–0.0850999	+21 PILOT
22	0.00490407	–10	54	–0.0908412	+22
23	–0.0407671	–9	55	–0.204513	+23
24	–0.0144953	–8	56	–0.344558	+24
25	–0.0103511	–7 PILOT	57	–0.617208	+25
26	–0.000136428	–6	58	–0.942483	+26
27	0.0116579	–5	59	0	+27
28	–0.00737179	–4	60	0	+28
29	0.0292066	–3	61	0	+29
30	0.0419534	–2	62	0	+30
31	0.0557991	–1	63	0	+31

6.7.4 Interpreting Spectral Flatness Data Summary

Standard	Total Sub-carriers	Data and Pilot Sub-carriers	Pilot Sub-carriers
802.11g	64 (–32 to +31)	52 (–26 to –1 & +1 to +26)	4 (–21, –7, +7 and +21)
802.11n (20 MHz)	64 (–32 to +31)	56 (–28 to –1 & +1 to +28)	4 (–21, –7, +7 and +21)
802.11n (40 MHz)	128 (–64 to +63)	114 (–58 to –2 & +2 to +58)	6 (–53, –25, –11, 11, 25, 53)
802.11ac (20 MHz)	64 (–32 to +31)	56 (–28 to –1 & +1 to +28)	4 (–21, –7, +7 and +21)
802.11ac (40 MHz)	128 (–64 to +63)	114 (–58 to –2 & +2 to +58)	6 (–53, –25, –11, +11, +25, +53)
802.11ac (80 MHz, 80+80 MHz)	256 (–128 to +127)	242 (–122 to –2 & +2 to +122)	8 (–103, –75, –39, –11, +11, +39, +75, +103)
802.11ac (160 MHz)	512 (–256 to +255)	484 (–250 to –130, –126 to –6 & +6 to 126 +130 to +250)	16 (–231, –203, –167, –139, –117, –89, –53, –25, +25, +53, +89, +117, +139, +167, +203, +231)
802.11ax (20 MHz)	256 (–128 to +127)	242 (–122 to –2 & +2 to +122)	8 (–116, –90, –48, –22, +22, +48, +90, +116)
802.11ax (40 MHz)	512 (–256 to +255)	484 (–244 to –3 & +3 to +244)	16 (–238, –212, –170, –144, –104, –78, –36, –10, +10, +36, +78, +104 +144, +170, +212, +238)
802.11ax (80 MHz)	1024 (–512 to +511)	996 (–500 to –3 & +3 to +500)	16 (–468, –400, –334, –266, –226, –158, –92, –24, +24, +92, +158 +226, +266, +334, +400, +468)

The “Total Subcarriers” column corresponds to the length of the spectral flatness array returned by FETC:SRW:SUMM:WLAN:OFDM:AEVM? Or FETC:SRW:PACK:WLAN:OFDM:AEVM?   For example, if the standard is 802.11ac (80 MHz) the spectral flatness array will have 256 elements, with index 0 corresponding to subcarrier –128.

To find the array index corresponding to a particular subcarrier, subtract the subcarrier number corresponding to index 0.

In the case of 802.11ac this is -128.

Example: Index for subcarrier -127 is –127 – (–128) = 1
Example: Index for pilot subcarrier +39 is +39 – (–128) = 167

The “Data & Pilot Subcarriers” column shows the positions of the modulated subcarriers. All other subcarriers are unmodulated. The value zero is returned in the spectral flatness array corresponding to those positions.

The “Pilot Subcarriers” column identifies the modulated subcarriers assigned to pilots.

6.7.5 Spectral Profile Measurement in 802.11g

Command	Description
CONF:SRW:SEGM:CLE	Clear any existing capture segments.
CONF:SRW:SEGM:APP WLG	Create a capture segment for WLAN 802.11g packets.
CONF:SRW:TIME 5E-03	Set the capture time to 5 ms Set a capture time sufficiently long to capture at least the number of packets required for the measurement.
CONF:SRW:FREQ 2.5E+09	Set the frequency to 2.5 GHz.
CONF:SRW:POW -10	Set the expected power level.
CONF:SRW:PACK 1	Set the number of packets to be measured.
CONF:SRW:WLG:DRAT 54MBPS	Set the expected data rate of the packets.
CONF:SRW:GATE:ENAB 1, ON CONF:SRW:GATE:TYPE 1,PACKET	Spectrum measurements require at least one gate to be set up. By setting the gate's type to 'PACKET' its start position and width will be set up automatically to cover the whole packet.
CONF:SRW:SEL:WLAN:EVM ON	Enable EVM measurements
CONF:SRW:SEL:WLAN:SPEC:NUM ON CONF:SRW:SEL:WLAN:SPEC:GRAP ON	Enable Spectrum Measurements, both numeric and graphical
Refer to Section 6.3	Set up the trigger
Refer to Section 6.4	Initiate the capture and wait for it to complete.
Refer to Section 7.1	Check that no errors occurred during the measurement.

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td>FETC:SRW:SUMM:WLAN:OFDM:EVM? 1</td><td>Read the numeric spectrum results
Example:
<table>
<tr><td>1</td><td>Segment index</td></tr>
<tr><td>0</td><td>Packet index (always zero for summary measurements.)</td></tr>
<tr><td>1</td><td>Number of gates</td></tr>
<tr><td>1</td><td>Start of results for gate number 1</td></tr>
<tr><td>0.0</td><td>RF Carrier suppression (DSSS only). Not relevant for an 802.11g packet.</td></tr>
<tr><td>TBD</td><td>Power spectral density</td></tr>
<tr><td>TBD</td><td>Occupied bandwidth</td></tr>
<tr><td>1</td><td>Spectrum mask passed limit test</td></tr>
<tr><td>0</td><td>RF Carrier suppression test passed (DSSS only). Not relevant for an 802.11g packet.</td></tr>
<tr><td>1</td><td>Power spectral density test passed</td></tr>
<tr><td>1</td><td>Occupied bandwidth test passed.</td></tr>
</table></td></tr>
<tr><td>:FETC:SRW:SUMM:WLAN:SPEC:GRAP?</td><td>Read the graphical spectrum results in default ASCII format
Example:
<table>
<tr><td>1</td><td>Segment index</td></tr>
<tr><td>0</td><td>Packet index (always zero for summary measurements).</td></tr>
<tr><td>1</td><td>Number of gates</td></tr>
<tr><td>1</td><td>Gate number of first gate.</td></tr>
<tr><td>70E+06</td><td>Bandwidth covered by spectrum (i.e. –35 MHz to +35 MHz)</td></tr>
<tr><td>2865</td><td>Number of points in spectrum</td></tr>
<tr><td>–75.12</td><td>Power level (dBm) of first point</td></tr>
<tr><td>…</td><td></td></tr>
<tr><td>–82.43</td><td>Power level (dBm) of last point.</td></tr>
</table></td></tr>
</table>

6.8 Continuous *Bluetooth* Measurements

Command	Description
`CONF:SRW:SEGM:CLE`	Clear any existing capture segments.
`CONF:SRW:SEGM:APP BT`	Create a capture segment for *Bluetooth* packets.
`CONF:SRW:TIME 5E-03`	Set the capture time to 5 ms. Set a capture time sufficiently long to capture at least the number of packets required for the measurement.
`CONF:SRW:FREQ 2.402E+09`	Set the frequency to 2.402 GHz.
`CONF:SRW:POW -10`	Set the expected power level.
`CONF:SRW:PACK 1`	Set the number of packets to be measured.
`CONF:SRW:BT:ADDR "000000C6967E"`	Set the *Bluetooth* address of the DUT. The address shown here is the setting used by default.
`CONF:SRW:BT:MODE STANDARD`	Set the *Bluetooth* measurement mode to 'SIG Standard'. This will cause the instrument to make measurements that are strictly in compliance with the *Bluetooth* SIG standard.
`CONF:SRW:BT:PACK DH1`	Set the packet type to be transmitted by the DUT.
`CONF:SRW:BT:PLEN 27`	Set the payload length expected from the DUT.
`CONF:SRW:BT:PAYL AAHEX`	Set the payload type expected from the DUT. In this case the pattern is 0xAA in hexadecimal, 10101010 in binary.

`CONF:SRW:SEL:BT:POW ON`
`CONF:SRW:SEL:BT:ICFT ON`
`CONF:SRW:SEL:BT:CDR ON`
`CONF:SRW:SEL:BT:MOD ON`
	Select all the numeric measurements applicable for a DH1 packet:
	• Transmit power
	• Initial carrier frequency tolerance
	• Carrier drift
	• Modulation characteristics

Refer to Section 6.3	Set up the trigger
Refer to Section 6.4	Initiate the capture and wait for it to complete.
Refer to Section 7.1	Check that no errors occurred during the measurement.

Command	Description
`FETC:SRW:PACK:BT:MOD? 1,1`	Read the modulation characteristics measurement results. Example:

1	Segment index
0	Packet index (always zero for summary measurements).
120E+03	Δf2 avg
128E+03	Δf2 max
NAN	Δf1 max (in SIG Standard mode this measurement cannot be made with a 10101010 payload)
NAN	Δf2 avg / Δf1 avg

100	%Δf2 max > 115 kHz
0	Δf1 avg lower limit test passed. (In SIG Standard mode this test cannot be performed with a 10101010 payload)
0	Δf1 avg upper limit test passed. (In SIG Standard mode this test cannot be performed with a 10101010 payload)
0	Δf2 avg / Δf1 avg limit test passed. (This test cannot be performed in SIG Standard mode)
1	%Δf2 max > 115 kHz limit test passed

Note:

For WLAN measurements, "NAN" means 'not a number' and is represented by the value 9.91E+37.

In SIG Compliant mode, some of the result fields are not available because the payload type is not as stipulated in the *Bluetooth* standard. In those cases the measurement returned is NAN and any associated limit check results are set to 'fail'.

In order to complete the SIG Compliant test of modulation characteristics, the DUT should be set up to generate packets with a payload of 0x0F (00001111 in binary) and the capture repeated.

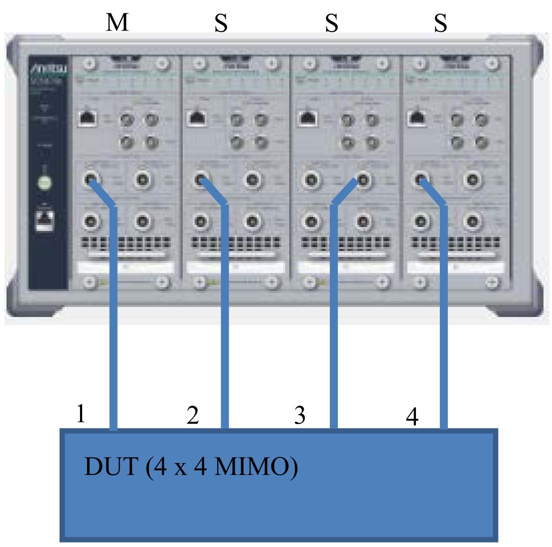
The results will then contain valid readings for Δf1 max and Δf1 avg (but not Δf2 avg and Δf2 max, which require the 10101010 payload). The ratio Δf2 avg / Δf1 avg must be performed in the user's program.

6.9 True MIMO Measurements

To perform a True MIMO transmitter measurement it is necessary to capture all spatial streams from the DUT simultaneously. True MIMO analysis includes cross-stream power leakage and its effect on error vector magnitude (EVM). This analysis depends on the ability to process measurement data from all the streams in one place.

MT8870A can be set up to use separate MU887000A modules to capture each spatial stream. The results are transferred to one "master" module for analysis.

Up to 8 spatial streams can be analyzed using up to 8 MU887000A modules.



Segments:

1 - TMIMOM "M"

2 - TMIMOS "S" (192.168.0.1)

3 - TMIMOS "S" (192.168.0.2)

4 - TMIMOS "S" (192.168.0.3)

Note that in this example stream 3 from the DUT is connected to **Port 4** of a MU887000A module rather than **Port 3**, which is used for the other streams. This is for illustrative purposes only and is explained in the text below.

One MU887000A, the "master" module, designated "M", is in overall charge of the measurement. All measurement set-up commands are sent to this module and it returns all the measurement results. The other modules, designated "S" are managed automatically by the "M" module.

The "M" module can be located in the slot 1 or slot 3 position of the MT8870A mainframe.

If "M" module is located in slot 1, up to seven "S" modules can be controlled by the "M" module and these can be located in any MT8870A mainframe.

The only constraints are:

1. The “M” module must be located in the slot 1 (left slot) position of its MT8870A mainframe.

2. All the MT8870A mainframes must be connected to the same local area network (LAN).

3. If any of the “S” modules are not located in the same mainframe as the “M” module, the **Trigger Input / Output 1** connectors on the rear panels of the MT8870A mainframes must be connected together.

If “M” module is located in slot 3, it can only control one “S” module, and this must be located in slot 4 of the same mainframe.

These conditions are needed to guarantee that data capture is synchronised between all modules used for the True MIMO measurement and that configuration and measurement data can be transferred between the “S” and “M” modules.

True MIMO measurements are managed using capture segments. A segment is defined for each spatial stream to be measured. After the measurement has completed, the summary or individual packet results for each MIMO stream can be fetched in the normal way, using the segment number as a parameter.

For example, the following command retrieves the EVM results for segment 2. In this case segment 2 maps to stream 2 of the DUT - see figure above.

```
:FETCh:SRW:SUMMary:WLAN:OFDM:EVM? 2
```

True MIMO capture segments are of two types: TMIMOM and TMIMOS.

TMIMOM is the segment type associated with the master “M” MU887000A module that controls the measurement. This segment is configured with the settings for the measurement.

TMIMOS is the segment type associated with the other MU887000A modules (the “S” modules) assigned to capturing other spatial streams. These segments hold limited set-up information. Only the RF port assignment, the power level (when autolevel is not used)- and the segment enable / disable setting can be changed. All other settings are defined by the TMIMOM segment and are set up on the “S” modules automatically.

True MIMO measurements are very similar to Sequential MIMO measurements in that a capture segment is defined for each spatial stream. The difference is that for Sequential MIMO, each segment corresponds to a portion of a capture on a single MU887000A module: namely the portion when the input port switch has selected the relevant spatial stream. For True MIMO, the segments correspond to simultaneous captures on two or more modules.

To set up a True MIMO measurement, use the following sequence of commands:

Command	Description
`:CONF:SRW:SEGM:CLE`	Clear any existing segments. It is not possible to mix True MIMO and other segment types in the same capture.
`:CONF:SRW:SEGM:APP TMIMOM`	Create a TMIMOM segment. This establishes the module as the “M” module for the True MIMO measurement.
`:CONF:SRW:SEGM:PORT PORT3`	Assign the RF input port for the “M” module’s measurements. It will also become the port assigned by default on the “S” modules when the TMIMOS segments are created. In this example the RF port is set to Port 3 on the TMIMOM segment (which controls the “M” module), and this setting will be set by default on the TMIMOS segments when they are created.

`:CONF:SRW:SEGM:APP TMIMOS`
`:CONF:SRW:SEGM:TMIM:IP “192.168.0.1”`
`:CONF:SRW:SEGM:APP TMIMOS`
`:CONF:SRW:SEGM:TMIM:IP “192.168.0.2”`
`:CONF:SRW:SEGM:APP TMIMOS`
`:CONF:SRW:SEGM:TMIM:IP “192.168.0.3”`
	Append a TMIMOS segment for each of the remaining spatial streams to be measured, and configure it with the IP address of the corresponding “S” module.
	In this example the DUT is a 4 x 4 MIMO device, so three TMIMOS segments are required corresponding to three “S” modules.

`:CONF:SRW:SEGM:ACT 1` `[Configuration commands…]`	Configure the TMIMOM segment. When the “M” module connects to the “S” modules it will configure them automatically with the same frequency and measurement settings.
`:CONF:SRW:SEGM:ACT 3` `:CONF:SRW:SEGM:PORT PORT4`	As shown in the diagram above, Port 4 (instead of the default Port 3) is used on the “S” module connected to stream 3 of the DUT.

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td><code>:CONF:SRW:TMIM:CONN</code>
<code>:STAT:SRW:TMIM:CONN? 0</code></td><td>The connect command causes the “M” module to attempt to connect to all the “S” modules.
The connection status query can be used to check the connections between the “M” and “S” modules.

It is not essential to use these commands, because the “M” module will connect to the “S” modules automatically in response to a measurement initiation or autolevel command. However they are useful when checking for connectivity problems and diagnosing them if they occur.

See <code>:STATus:SRWireless:TMIMo:CONNection?</code> for details of the response.</td></tr>
<tr><td><code>:CONF:SRW:TMIM:DISC</code></td><td>This command forces the “M” module to disconnect from the “S” modules.
It is used to release the “S” modules so that they may be controlled independently.
Disconnection from the “S” modules also occurs automatically under the following circumstances:
The connection to the “M” module is closed.
All segments are cleared.
The “TMIMOM” segment is deleted.</td></tr>
<tr><td><code>:INIT:SRW:ALEV</code></td><td>This command causes the following actions to occur:
The “M” module connects to the “S” modules if necessary.
The “M” module configures the “S” modules.
The “M” module performs an autolevel operation and simultaneously initiates autolevel operations on the connected “S” modules.</td></tr>
<tr><td><code>:INIT:SRW</code></td><td>This command causes the following actions to occur:
The “M” module connects to the “S” modules if necessary.
The “M” module configures the “S” modules.
The “M” module arms its own trigger and that of the “S” modules.
When the “M” module is triggered it triggers the “S” modules simultaneously to initiate synchronised captures on all modules.
On completion of the capture, the “M” module reads and processes the measurement data from the “S” modules.

On completion, the measurement results may be fetched from the “M” module in the normal way.</td></tr>
</table>

6.10 Summary Results vs. Packet Results

The SRW application can return measurement results for individual packets or summary results covering a number of packets. The table below compares summary and packet EVM results for an 802.11g signal.

	Summary Results		Packet Results (Packet 1)	
Field	**Results**	**Description**	**Result**	**Description**
0	1	Segment index	1	Segment index
1	0	Packet index (always zero for summary measurements.)	1	Packet index
2	64	Number of subcarriers	64	Number of subcarriers
3	38	Number of symbols	38	Number of symbols
4	–39.6003	Centre frequency leakage	–39.6003	Centre frequency leakage
5	–0.23	Symbol clock frequency tolerance (ppm)	–0.23	Symbol clock frequency tolerance (ppm)
6	0.124969	Centre frequency tolerance (ppm)	0.124969	Centre frequency tolerance (ppm)
7	300.176	Centre frequency tolerance (Hz)	300.176	Centre frequency tolerance (Hz)
8	–38.2331	Max EVM in segment (dB)	–38.2331	Max EVM (dB)
9	1.22559	Max EVM in segment (%)	1.22559	Max EVM (%)
10	–62.579	Min EVM (dB)	–62.579	Min EVM (dB)
11	0.0744916	Min EVM (%)	0.0744916	Min EVM (%)
12	–45.7253	Average EVM (dB)	–45.7253	Average EVM (dB)
13	0.517289	Average EVM (%)	0.517289	Average EVM (%)
76	1	Centre frequency leakage test pass flag	1	Centre frequency leakage test pass flag
77	1	Symbol clock frequency tolerance test pass flag	1	Symbol clock frequency tolerance test pass flag
78	1	Centre frequency leakage test pass flag	1	Centre frequency leakage test pass flag
79	1	EVM test pass flag	1	EVM test pass flag
80	1	Spectral flatness test pass flag	1	Spectral flatness test pass flag
81	1	Packet number for max EVM		
82	2	Packet number for min EVM		

Summary results apply to the whole capture; packet results apply to an individual packet.

The table shows that, for convenience, the order of fields in a summary measurement matches that of a packet measurement. If your program splits the results fields into an array, the same index can be used to retrieve either the result from an individual packet measurement or the related average result from a summary measurement.

Commands to retrieve summary measurements begin FETCh:SRW:SUMMary.

Commands to retrieve individual packet results begin FETCh:SRW:PACKet…

Summary measurement results contain additional fields to identify 'best case' and 'worst case' packets in the capture. For example, the EVM measurement summary results include the packet numbers of the packets with the worst and best EVM.

Note that in this example we have opted to analyze only one packet. This means that the summary results and packet results are identical.

Notes:

For WLAN measurements, the limit checking flags in summary results are set by comparing the average results for the segment against the limits.

For *Bluetooth* measurements, the limit checking flags are the logical AND of the individual packet results. In other words, each individual packet must meet specification. This is a requirement of the *Bluetooth* SIG standard

6.11 IEEE 802.11ac 80+80 Measurements

Introduction

In addition to 160 MHz (VHT160) transmissions, the 802.11ac standard specifies a method for transmitting a pair of 80 MHz *frequency segments* that can be positioned separately within the 5 GHz WLAN band to minimize the effect of interference from other services. This type of transmission is known as "80+80".

Within the IEEE standard, the 80+80 frequency segments are referred to as "primary" and "secondary". The use of "primary" and "secondary" does not describe which frequency segment is located at the higher frequency. The secondary frequency segment can be at a lower or higher frequency than the primary.

For clarity, this manual uses the terms "lower" and "upper" to distinguish between the two carrier frequencies.

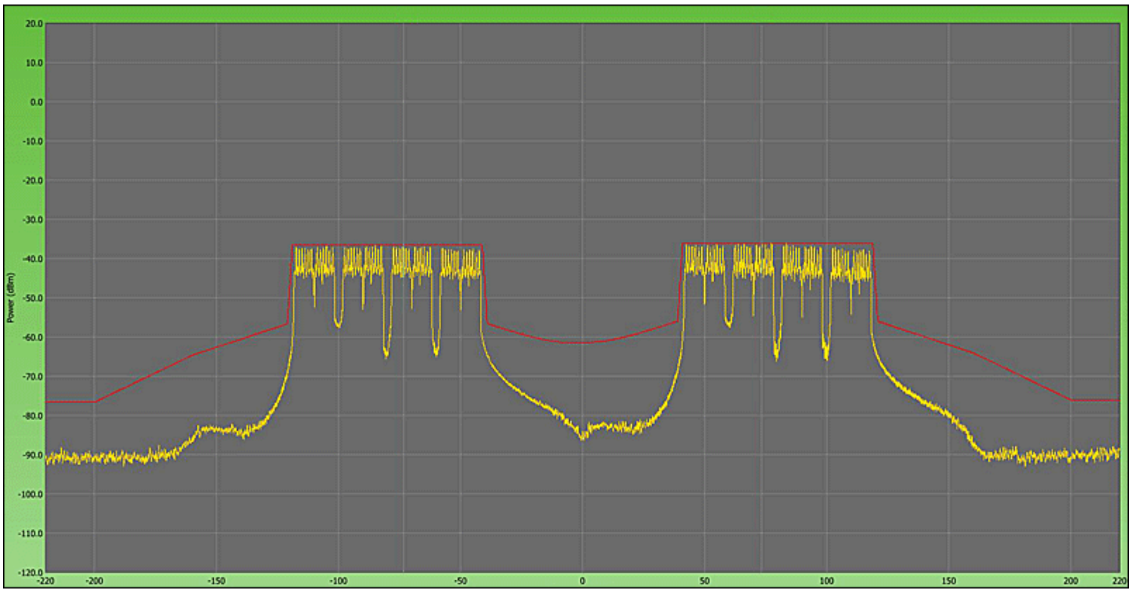


Figure 6.11-1 – 80+80 Transmission

Capture Segments for 802.11ac 80+80 Measurements

The Short Range Wireless Application can measure the following 80+80 signals:

- 80+80 SISO concurrently or sequentially. Concurrent measurements require two MU887000A modules, but guarantee to capture the 80+80 upper and lower frequency segments from the same packet.

Sequential measurements require a single MU887000A. The upper and lower frequency segments of the 80+80 are captured sequentially.

- 80+80 MIMO signals either fully concurrently, or using a combination of concurrent and sequential measurements to reduce the number of MU887000A modules required.

The measurement results include the combined EVM of the two 80+80 frequency segments and spectral emission measurements against a combined mask calculated according to the IEEE standard.

A number of different capture segment types are available to allow you to set up these measurements.

SEGMENT TYPE	PURPOSE
SEQ80_80SISO	Used to make sequential 80+80 SISO measurements with a single MU887000A module. You need to set up two of these capture segments: one for each carrier.
CON80_80SISOM CON80_80SISOS	Used to make concurrent 80+80 SISO measurements using two MU887000A modules. To use these segment types you need a MT8870A mainframe containing at least two MU887000A modules. You set up the measurement on the module in the slot 1 position and it controls a second module in one of the other slots to make a synchronized concurrent measurement. ***Note:*** Alternatively you can set up the measurement on a module in the slot 3 position, but in that case the module it controls must always be in the slot 4 position. If four MU887000A modules are installed in the MT8870A mainframe, it is possible to make concurrent SISO measurements on two DUTs simultaneously. The "M" and "S" suffixes indicate "Master" and "Slave" control using a similar convention to capture segments for capturing true MIMO signals. In this case you need to define one CON80_80SISOM segment followed by one CON80_80SISOS segment. The first segment controls the master module on which you set up the measurement and from which you read the results, and the second segment controls the "slave" module automatically.

SEGMENT TYPE	PURPOSE
CON80_80TMIMOM CON80_80TMIMOS	Used to make concurrent 80+80 true MIMO measurements. The number of MU887000A modules you need to make this measurement is given by the formula: $Number\ of\ 80+80\ Segments * Number\ of\ space-time\ streams$ A 2 x 2 MIMO device that supports 80+80 will therefore require a total of four modules (two frequency segments multiplied by two space-time streams). You set up the measurement on the module in the slot 1 position and it controls the remaining modules to make a synchronized concurrent measurement. The "M" and "S" suffixes indicate "Master" and "Slave" control using a similar convention to capture segments for capturing true MIMO signals. In this case you need to define one CON80_80TMIMOM capture segment followed by a CON80_80TMIMOS capture segment for each of the other modules used for the measurement. The first segment controls the master module on which you set up the measurement and from which you read the results, and the remaining segments control the "slave" modules automatically.
SEQ80_80TMIMOM SEQ80_80TMIMOS	You can use these segment types to make true MIMO measurements with half the number of MU887000A modules needed for a fully concurrent 80+80 true MIMO measurement. All the spatial streams on one of the two 80+80 carriers are captured concurrently from the same packet. Then all the spatial streams on the other 80+80 carrier are captured concurrently from a different packet. You need to use a separate MU887000A module for each spatial stream. A 2 x 2 MIMO device that supports 80+80 will therefore require two modules. You set up the measurement on the module in the slot 1 position and it controls the second module to make a synchronized concurrent measurement. The "M" and "S" suffixes indicate "Master" and "Slave" control using a similar convention to capture segments for capturing true MIMO signals. In this case you need to define two SEQ80_80TMIMOM capture segments, one after the other. The first controls synchronized measurements of all spatial streams for the lower frequency segment of the 80+80 transmission, and the second does the same for the upper frequency of the 80+80 transmission. Following these two master segments, you need to define a pair of SEQ80_80TMIMOS capture segments for each remaining spatial stream. The first segment of each pair is set to measure at the lower frequency, and the second to measure at the upper frequency. The master segments control the master module on which you set up the measurement and from which you read the results, and the remaining pairs of "S" segments control the slave module automatically to measure the remaining spatial streams.

Combined and Separate 80+80 Signals

The two frequency segments forming an 80+80 transmission are normally combined on the same antenna port, but some first generation devices used separate antenna ports. There is a configuration command to specify whether the 80+80 signals are combined within the DUT or output separately:

```
:CONFigure:SRWireless:SPECtrum:C80_80
```

The default setting is `COMBINED`, which is correct for most DUTs, however if the 80+80 frequency segments are output on separate ports, you must change this setting to `SEPARATE` to ensure that the instrument calculates the combined 80+80 spectrum correctly.

“Full Span” Spectrum Measurements are Always Enabled

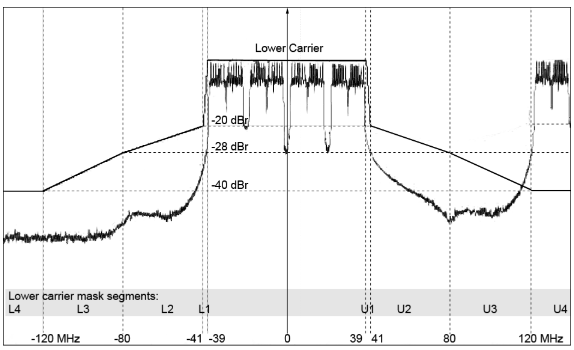
When measuring the spectral profile of 80+80 signals it is necessary to take into account the contribution to the measurement of the other carrier because this can change the shape of the spectral mask and affect mask margin measurements. For this reason, “full span” spectrum measurements (±140 MHz span) are enabled automatically when measuring 80+80 transmissions.

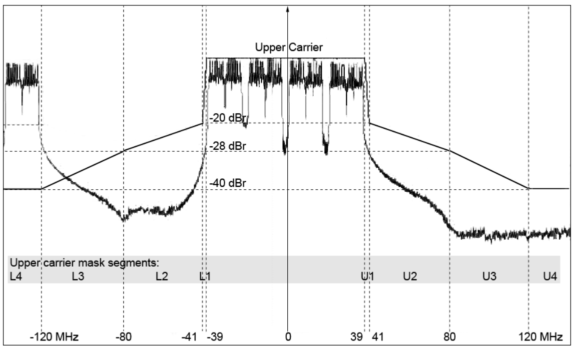
Minimum Separation of the Lower and Upper Frequency Segments

The minimum separation of the two carriers is 80 MHz.
An error is generated if you try to set a smaller frequency separation.
The reason for this is that the two signals will interfere with each other if the separation is less than their individual bandwidths (80 MHz).

Calculating the Spectral Mask

When measuring the spectral profile of 80+80 signals, the instrument calculates a combined spectral mask according to the IEEE 802.11ac standard. The shape of the combined mask depends on the frequency separation of the lower and upper frequency segments and their relative power levels.





The two spectrum displays above show measurements centered about the lower and upper carriers. For illustration, the standard IEEE spectral mask of a VHT80 signal is superimposed in each case. However, as you can see, the presence of the other carrier means that the standard mask is not suitable for this measurement. A combined mask must be calculated.

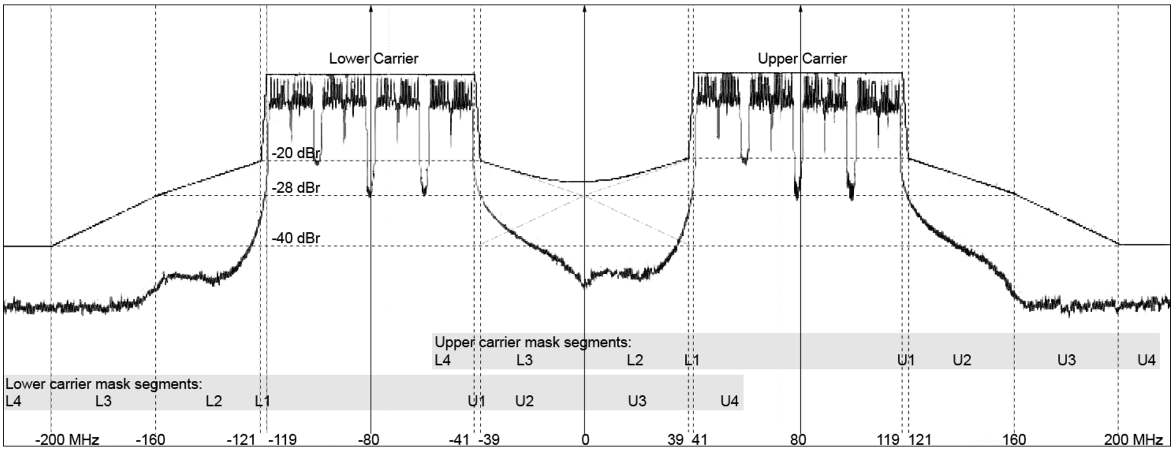


Figure 6.11-2 - 80+80 Combined Spectral Mask

As Figure 6.11-2 shows, the mask in the region between the two carriers is calculated according to the IEEE standard, which takes into account the frequency separation and the relative power level.

The commands `:FETC:SRW:SUMM:WLAN:SPEC:NUM?` and `:FETC:SRW:SUMM:WLAN:SPEC:NUM?` are used to retrieve numeric spectrum results for each frequency segment separately. All mask margin and limit test results returned by these commands are calculated against the combined mask. As Figure 6.11-2 shows, the existing mask segment labels (L1 to L4 and U1 to U4) are used for the combined mask case. You can retrieve the mask margin results in the region between the two carriers by fetching the numeric spectrum results for the lower carrier and checking mask segments U1 to U4, or instead by fetching the results for the upper carrier and checking mask segments L1 to L4.

6.11.1 802.11ac 80+80 SISO Sequential Example – Single Module

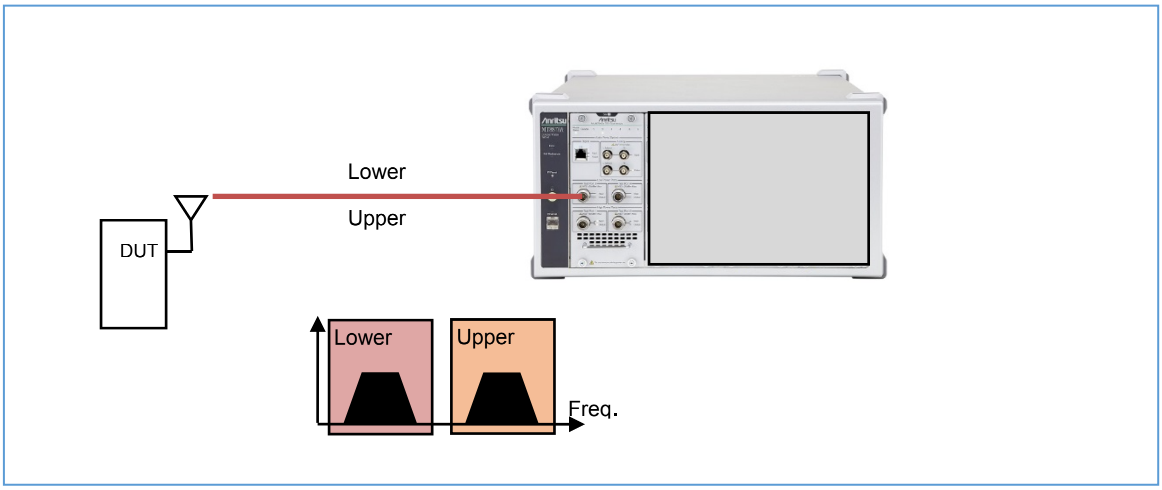


Figure 6.11.1-1 - 80+80 Signals Combined in DUT

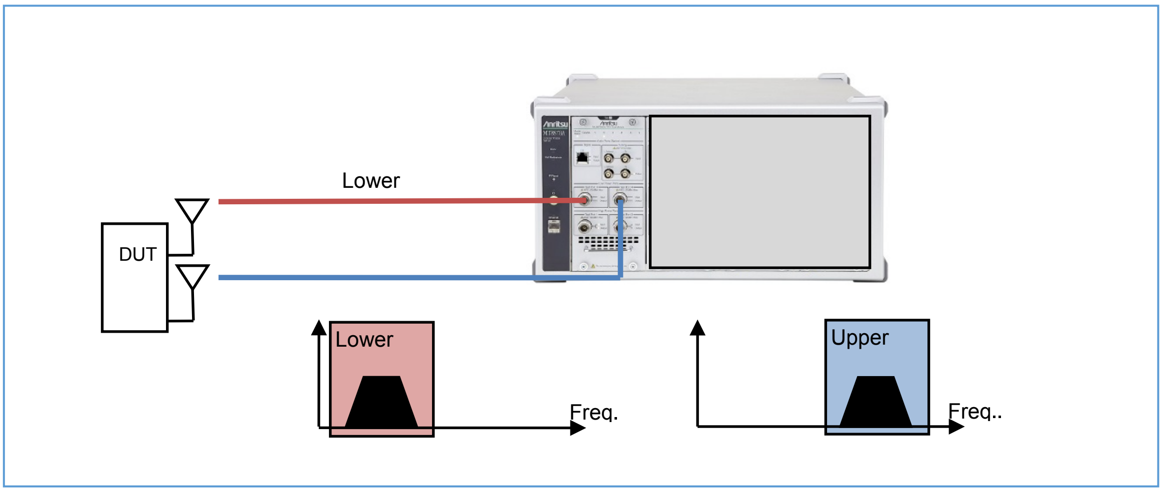


Figure 6.11.1-2 - 80+80 Signals on Separate DUT Ports

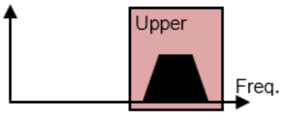
"Figure 6.11.1-1 - 80+80 Signals Combined in DUT" shows the configuration for measuring an 80+80 transmission using a single MU87000A module for the case when the lower and upper frequency segments are combined on a single antenna port. "Figure 6.11.1-2 - 80+80 Signals on Separate DUT Ports" shows the set-up when the lower and upper frequency segments are transmitted from separate antenna ports.

The example below shows how to capture and measure an 80+80 SISO transmission with the following characteristics:

- Lower frequency segment: 5290 MHz
- Upper frequency segment: 5610 MHz

- Frequency segments are combined within the DUT. (See Figure 6.11.1-1 for connection diagram)
- Power level will be set up using Autolevel

Relationship between capture segments measurements and time:

	Time →	
	First Capture	**Second Capture**
Segment 1	Measure lower carrier: 5290 MHz 	
Segment 2		Measure upper carrier: 5610 MHz 

Command	Description
	[Preliminaries when controlling a MU88700x]
`SYST:LANG SCPI`	Switch to SCPI mode ***Note:*** Do not use a colon character, ":", before this command.
`:INST SRW`	Select the SRW Application (WLAN / Bluetooth)
`:CALC:CAL:BAND:STAR:TEMP 2.0`	Run a calibration. In this case the calibration will only run if the temperature inside the instrument has changed by 2.0 degrees since the previous calibration.
`*WAI`	Wait for the calibration to complete. It takes few minutes to finish the calibration.
`:CALC:CAL:BAND:RES?`	Check the calibration was successful. The response should be **PASS**.
	[Set up two 80+80 SISO capture segments, one for each frequency.]
`:CONF:SRW:SEGM:CLE`	Clear any existing capture segments.
	[Segment 1 will capture the 80+80 lower frequency transmission]
`:CONF:SRW:SEGM:APP SEQ80_80SISO`	Create a capture segment for sequential 80+80.

Command	Description
:CONF:SRW:SEGM:ACT 1	Make segment 1 the active segment. ***Note:*** Strictly this is not necessary because the most recently added segment always becomes the active segment. However it is good practice to set the active segment explicitly when working with multiple segments.
[:CONF:SRW:WLAC:SPEC:C80_80 COMBINED]	Specify whether the 80+80 signals are combined in the DUT or output on separate antenna ports. ***Note 1:*** Use SEPARATE if the primary and secondary 80+80 frequency segments are output from the DUT on separate antenna ports and connected to separate RF ports on the MU887000A module. If the signals are combined either in the DUT or externally using a combiner, set the value of this setting to COMBINED. The default value of this setting is COMBINED, so for this example the command is optional.
:CONF:SRW:SEGM:PORT PORT3	Set the port. In this example the port chosen is 3.
:CONF:SRW:FREQ 5.29E+09	Set the frequency of the lower frequency segment to 5290 MHz.
:CONF:SRW:CAPT:MODE PACKET	Set the capture mode to Packet Count.
:CONF:SRW:PACK 10	Set the number of packets to be measured.
:CONF:SRW:TRIG IMMEDIATE	For this example the DUT will be set up to generate packets continuously. For that reason it is simpler to let the system locate and capture packets automatically by selecting IMMEDIATE triggering.
	[Segment 2 will capture the 80+80 upper frequency segment]
:CONF:SRW:SEGM:APP SEQ80_80SISO	Create a capture segment for sequential 80+80.
:CONF:SRW:SEGM:ACT 2	Make segment 2 the active segment.
:CONF:SRW:PORT PORT3	Set the port. Because the 80+80 frequency segments are combined, the port number is the same as segment 1.
:CONF:SRW:FREQ 5.61E+09	Set the frequency to 5610 MHz, the upper frequency of the 80+80 transmission.

Command	Description
	[Setting up measurements]
`:CONF:SRW:SEGM:ACT 1`	Make the first segment the active segment. Any measurement settings on this segment will be applied automatically to segment 2.
`:CONF:SRW:SEL:WLAN:EVM:COMB ON` `:CONF:SRW:SEL:WLAN:SPEC:GRAP ON`	SEQ80_80SISO segments have the following measurements enabled by default: • Power • EVM (Basic) • Spectrum (Numeric) Use :CONF:SRW:SEL… commands to enable and disable alternative sets of measurements. In this example we will enable combined EVM measurements and graphical spectrum measurements. ***Note:*** Any measurement enabled on the first segment of a pair of `SEQ80_80SISO` segments is automatically enabled on the other segment of that pair. So in this example you need only enable measurements on segment 1. It is not necessary to enable "full span" spectrum measurements for 80+80 signals. This feature is enabled automatically.
	[Autolevel]
`:INIT:SRW:ALEV`	Using the DUT manufacturer's control software, set up the DUT to transmit 80+80 packets continuously, then: Initiate Autolevel
`*WAI`	Wait for the operation to complete

Command	Description
:CONF:SRW:POW?	Read back the power level set by the autolevel function. You can use this response to confirm that the system is working as expected. For example, if the DUT typically generates a signal at +17 dBm, but the autolevel function returns –60 dBm, the reason could be that: • the wrong port has been selected for measurement on the MU887000A module; or • the DUT has not been set up correctly; or • the DUT is not connected to the port selected for measurement.
	[Capture]
:INIT:SRW	Initiate the capture.
*WAI	Wait for the operation to complete.
:STAT:SRW:MEAS?	Read the measurement status. It should be **1**, which indicates measurement complete.
	[Check Capture Information]
Refer to Section 7.1	Check that no errors occurred during the measurement.
	[Retrieve Measurements]
:FETC:SRW:SUMM:WLAN:OFDM:EVM? 1,0 :FETC:SRW:SUMM:WLAN:OFDM:EVM? 2,0 :FETC:SRW:SUMM:WLAN:OFDM:EVM:COMB? 1,0	Read the EVM measurement summary results for each segment. (The first command parameter is the segment number.) In this case the result from segment 1 will be the EVM of the lower frequency segment and that from segment 2 will be the EVM of the upper frequency segment. The combined EVM result can be fetched from either segment. In this example it is fetched from segment 1.

6.11.2 802.11ac 80+80 SISO Concurrent Example – Two Modules

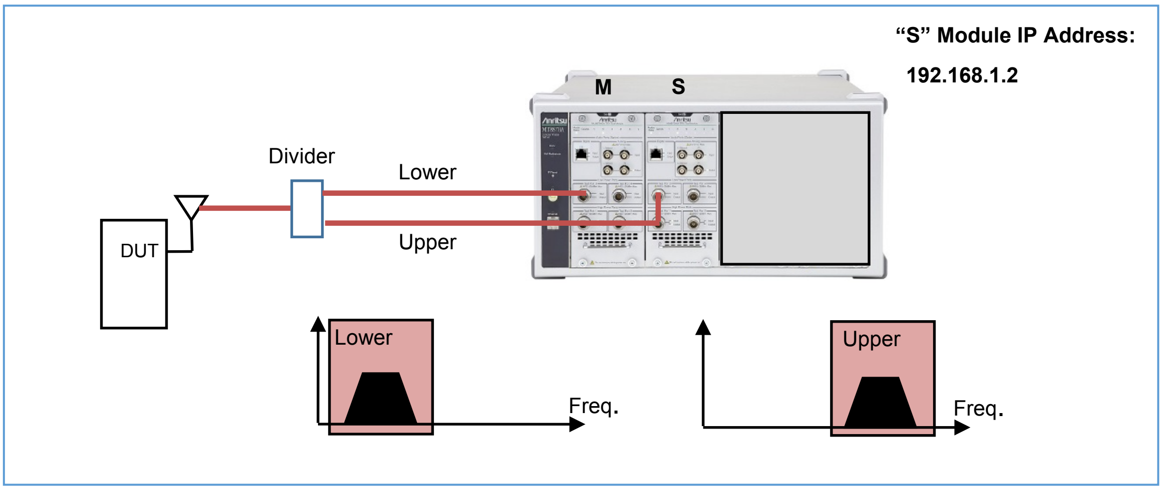


Figure 6.11.2-1 - 80+80 Signals Combined in DUT

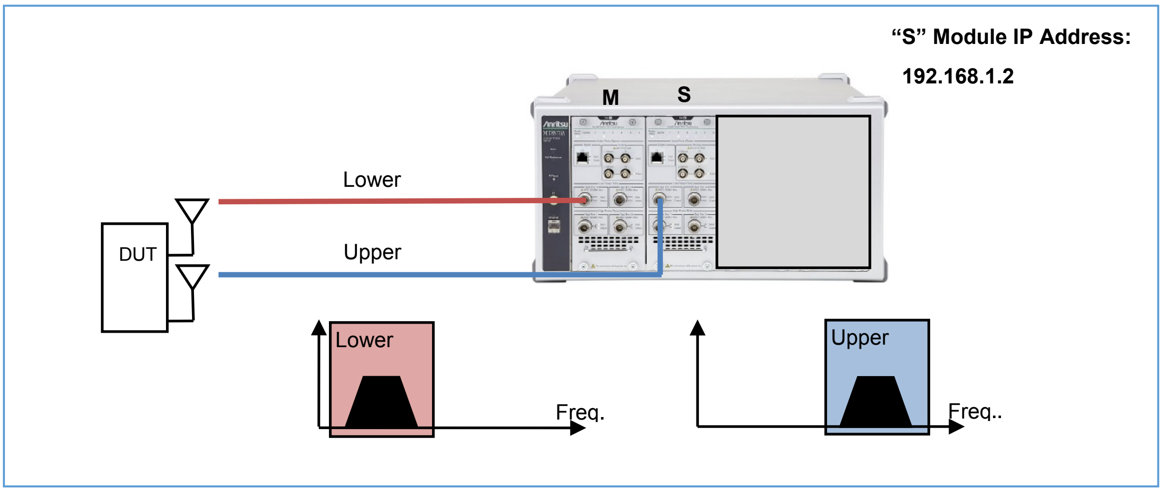


Figure 6.11.2-2 - 80+80 Signals on Separate DUT Ports

"Figure 6.11.2-1 - 80+80 Signals Combined in DUT" shows the configuration for measuring an 80+80 transmission using two MU87000A modules when both the lower and upper frequency segments are combined on a single antenna port. "Figure 6.11.2-2 - 80+80 Signals on Separate DUT Ports" shows the set-up when the lower and upper frequency segments are transmitted from separate antenna ports.

One MU887000A, the "master" module, designated "M", is in overall charge of the measurement. All measurement set-up commands are sent to this module and it returns all the measurement results. The other module, designated "S" is managed automatically by the "M" module.

Notes:

- The “M” module must be located in the slot 1 position of the MT8870A mainframe (as shown above), or in the slot 3 position.

- If the “M” module is located in the slot 3 position, the “S” module that it controls must be in slot 4.

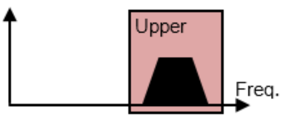
- If the “M” module is located in the slot 1 position, the “S” module can be in any of the other slots, or even in a separate MT8870A mainframe provided that MT8870A mainframe is connected to the same local area network (LAN) and the **Trigger Input / Output 1** connectors on the rear panels of the mainframes are connected together.

- You need to know the IP address of the “S” module so that the “M” module can be set up to control it. In this example the IP address used is 192.168.1.2. Substitute the correct IP address for your “S” module.

The following example shows how to capture and measure an 80+80 SISO signal with the following characteristics:

- Lower frequency segment: 5290 MHz
- Upper frequency segment: 5610 MHz
- Frequency segments are combined within the DUT. (See“Figure 6.11.2-1 - 80+80 Signals Combined in DUT” for connection diagram)
- Power level will be set up using Autolevel

Relationship between capture segments measurements and time:

	Time →	
	First Capture	**Second Capture**
Segment 1	Measure lower carrier: 5290 MHz on “M” module 	Not Applicable (Measurements are concurrent.)
Segment 2	Measure upper carrier: 5610 MHz on “S” module 	

All the following commands are sent to the MT887000A module designated as the “M” module:

Command	Description
	[Preliminaries when controlling a MU88700x]
`SYST:LANG SCPI`	Switch to SCPI mode ***Note:*** Do not use a colon character, “:”, before this command.
`:INST SRW`	Select the SRW Application (WLAN / Bluetooth)
`:CALC:CAL:BAND:STAR:TEMP 2.0`	Run a calibration. In this case the calibration will only run if the temperature inside the instrument has changed by 2.0 degrees since the previous calibration.
`*WAI`	Wait for the calibration to complete. It takes few minutes to finish the calibration.
`:CALC:CAL:BAND:RES?`	Check the calibration was successful. The response should be **PASS**.
	[Set up two concurrent 80+80 SISO capture segments, one for each frequency.]
`:CONF:SRW:SEGM:CLE`	Clear any existing capture segments.
	[Segment 1 will capture the 80+80 lower frequency transmission]
`:CONF:SRW:SEGM:APP CON80_80SISOM`	Create a capture segment for concurrent 80+80 SISO measurements. This establishes the module as the “M” module for the measurement.
`:CONF:SRW:SEGM:ACT 1`	Make segment 1 the active segment. ***Note:*** Strictly this is not necessary because the most recently added segment always becomes the active segment. However it is good practice to set the active segment explicitly when working with multiple segments.

Command	Description
[:CONF:SRW:WLAC:SPEC:C80_80 COMBINED]	Specify whether the 80+80 upper and lower frequency segments are combined in the DUT or output on separate antenna ports. ***Note 1:*** Use SEPARATE if the upper and lower 80+80 frequency segments are output from the DUT on separate antenna ports. If the signals are combined either in the DUT or externally using a combiner, set the value of this setting to COMBINED. The default value of this setting is COMBINED, so for this example the command is optional.
:CONF:SRW:SEGM:PORT PORT3	Assign the RF input port for the "M" module's measurements. It will also become the port assigned by default on the "S" module when the CON80_80SISOS segment is created. In this example we are using port 3.
:CONF:SRW:FREQ 5.29E+09	Set the frequency of the lower 80+80 frequency segment to 5290 MHz.
:CONF:SRW:CAPT:MODE PACKET	Set the capture mode to Packet Count.
:CONF:SRW:PACK 10	Set the number of packets to be measured.
:CONF:SRW:TRIG IMMEDIATE	For this example the DUT will be set up to generate packets continuously. For that reason it is simpler to let the system locate and capture packets automatically by selecting IMMEDIATE triggering.
	[Segment 2 will capture the 80+80 upper frequency segment concurrently using a separate module]
:CONF:SRW:SEGM:APP CON80_80SISOS	Create a capture segment for concurrent 80+80 SISO measurements. The "S" suffix means that this segment will control the "S" or "slave" MU887000A module to capture the other 80+80 frequency segment.
:CONF:SRW:SEGM:ACT 2	Make segment 2 the active segment.
:CONF:SRW:SEGM:PORT PORT3	Set the port.
:CONF:SRW:FREQ 5.61E+09	Set the frequency to 5610 MHz, the upper frequency of the 80+80 transmission.

Command	Description
`:CONF:SRW:SEGM:TMIM:IP "192.168.1.2"`	Set the IP address of the "S" module. (This command is the same as is used for True MIMO measurements)
	[Setting up measurements]
`:CONF:SRW:SEGM:ACT 1`	Make the first segment the active segment. Any measurement settings on this segment will be applied automatically to segment 2.
`:CONF:SRW:SEL:WLAN:EVM:COMB ON` `:CONF:SRW:SEL:WLAN:SPEC:GRAP ON`	CON80_80SISOM and CON80_80SISOS segments have the following measurements enabled by default: • Power • EVM (Basic) • Spectrum (Numeric) Use :CONF:SRW:SEL... commands to enable and disable alternative sets of measurements. In this example we will enable combined EVM measurements and graphical spectrum measurements. ***Note:*** Any measurement enabled on the master segment is automatically enabled on the "S" segment. So in this example you need only enable measurements on segment 1. It is not necessary to enable "full span" spectrum measurements for 80+80 signals. This feature is enabled automatically.
	[Autolevel]
`:INIT:SRW:ALEV`	Using the DUT manufacturer's control software, set up the DUT to transmit 80+80 packets continuously, then: Initiate Autolevel
`*WAI`	Wait for the operation to complete

Command	Description
:CONF:SRW:POW? :CONF:SRW:SEGM:ACT 2 :CONF:SRW:POW? :CONF:SRW:SEGM:ACT 1 <- See note.	This sequence of commands reads back the power set by the autolevel function on each module. You can use this response to confirm that the system is working as expected. For example, if the DUT typically generates a signal at +17 dBm, but the autolevel function returns –60 dBm, the reason could be that: • the wrong port has been selected for measurement on one of the MU887000A modules; or • the DUT has not been set up correctly; or • the DUT is not connected to the port selected for measurement. ***Note:*** You must set the active segment back to 1 before initiating the capture.
	[Capture]
:INIT:SRW	Initiate the capture.
*WAI	Wait for the operation to complete.
:STAT:SRW:MEAS?	Read the measurement status. It should be **1**, which indicates measurement complete.
	[Check Capture Information]
Refer to Section 7.1	Check that no errors occurred during the measurement.
	[Retrieve Measurements]
:FETC:SRW:SUMM:WLAN:OFDM:EVM? 1,0 :FETC:SRW:SUMM:WLAN:OFDM:EVM? 2,0 :FETC:SRW:SUMM:WLAN:OFDM:EVM:COMB? 1,0	Read the EVM measurement summary results for each segment. (The first command parameter is the segment number.) In this case the result from segment 1 will be the EVM of the lower frequency segment and that from segment 2 will be the EVM of the upper frequency segment. The combined EVM result can be fetched from either segment. In this example it is fetched from segment 1.

6.11.3 802.11ac Concurrent 80+80 TMIMO Example – Four Modules

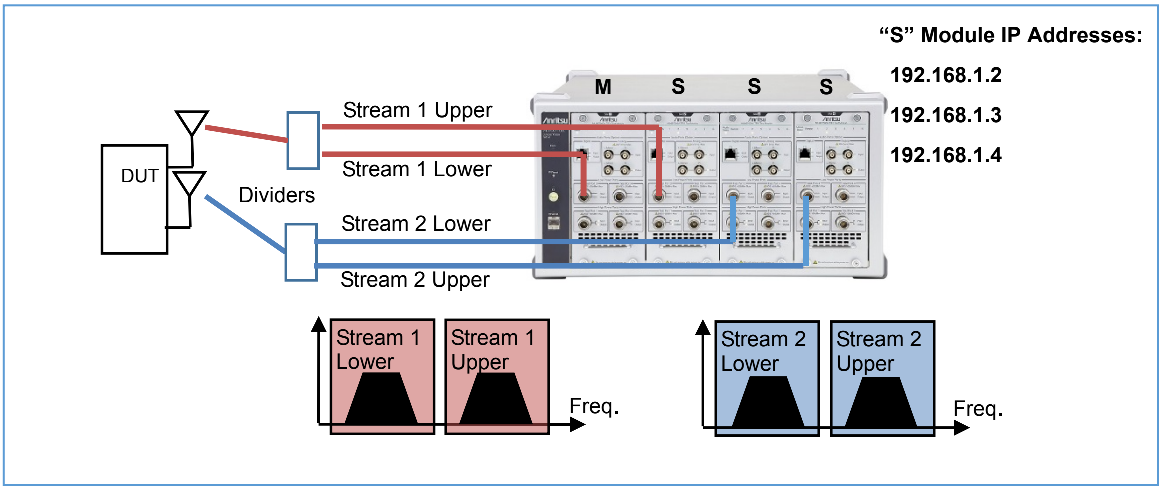


Figure 6.11.3-1 – 2x2 MIMO – 80+80 Signals Combined in DUT

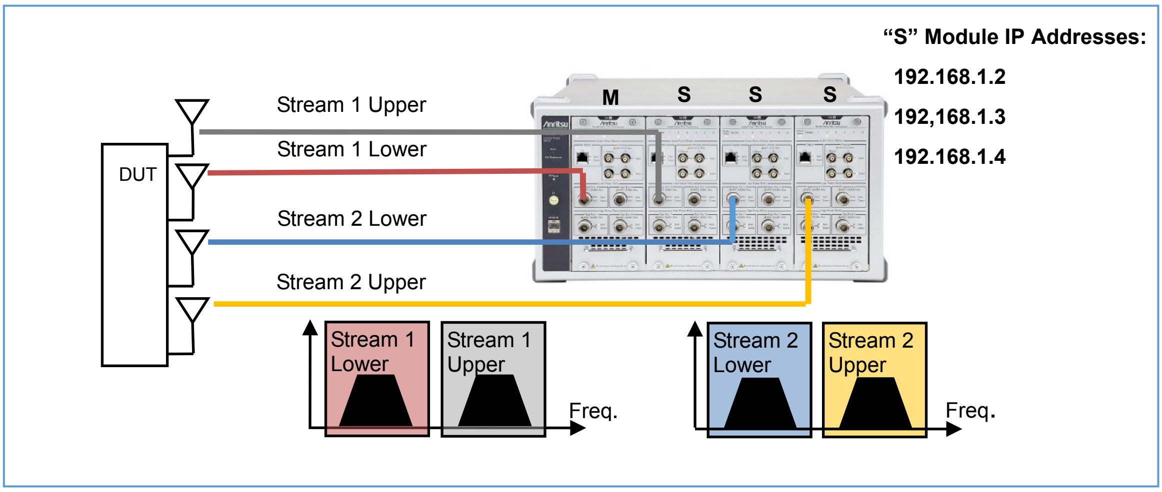


Figure 6.11.3-2 – 2x2 MIMO – 80+80 Signals on Separate DUT Ports

"Figure 6.11.3-1 – 2x2 MIMO – 80+80 Signals Combined in DUT" shows the configuration for measuring a 2x2 MIMO 80+80 transmission using four MU87000A modules. There are two antenna ports, each carrying one of the two space-time streams. Each space-time stream is carried by an 80+80 signal.

"Figure 6.11.3-2 – 2x2 MIMO – 80+80 Signals on Separate DUT Ports" shows the set-up when the lower and upper 80+80 frequency segments forming each space-time stream are transmitted from separate antenna ports.

One MU887000A, the "master" module, designated "M", is in overall charge of the measurement. All measurement set-up commands are sent to this module and it returns all the measurement results. The other module, designated “S” is managed automatically by the “M” module.

Notes:

- The “M” module must be located in the slot 1 position of the MT8870A mainframe.

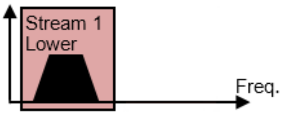
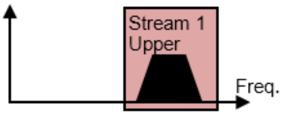
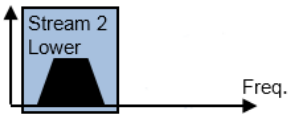
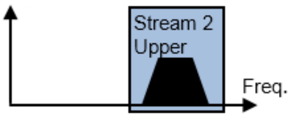
- The “S” modules can be in any of the other slots of the same mainframe, or in a separate MT8870A mainframe provided that the second MT8870A is connected to the same local area network (LAN) and the **Trigger Input / Output 1** connectors on the rear panels of the mainframes are connected together.

- You need to know the IP addresses of the “S” modules so that the “M” module can be set up to control them. In this example the IP addresses used are 192.168.1.2 to 192.168.1.4. Substitute the correct IP addresses for your “S” modules.

The following example shows how to capture and measure an 80+80 2x2 MIMO signal with the following characteristics:

- Lower frequency segment: 5290 MHz
- Upper frequency segment: 5610 MHz
- Frequency segments are combined within the DUT. (See Figure 6.11.3-1 for connection diagram)
- Power level will be set up using Autolevel

Relationship between capture segments, measurements and time:

	Time →	
	First Capture	**Second Capture**
Segment 1	Measure space-time stream 1 on lower carrier: 5290 MHz on "M" module  	
Segment 2	Measure space-time stream 1 on upper carrier: 5610 MHz on first "S" module  	
Segment 3	Measure space-time stream 2 on lower carrier: 5290 MHz on second "S" module  	Not Applicable (Measurements are concurrent)
Segment 4	Measure space-time stream 2 on upper carrier: 5610 MHz on third "S" module  	

All commands are sent to the MT887000A module designated as the "M" module:

Command	**Description**
	[Preliminaries when controlling a MU88700x]
`SYST:LANG SCPI`	Switch to SCPI mode ***Note:*** Do not use a colon character, ":", before this command.
`:INST SRW`	Select the SRW Application (WLAN / Bluetooth)
`:CALC:CAL:BAND:STAR:TEMP 2.0`	Run a calibration. In this case the calibration will only run if the temperature inside the instrument has changed by 2.0 degrees since the previous calibration.
`*WAI`	Wait for the calibration to complete. It takes few minutes to finish the calibration.
`:CALC:CAL:BAND:RES?`	Check the calibration was successful. The response should be **PASS**.

Command	Description
	[Set up four concurrent 80+80 T-MIMO capture segments: **2 80+80 frequency segments x 2 space-time streams.]**
`:CONF:SRW:SEGM:CLE`	Clear any existing capture segments.
	[Segment 1 is the master segment that will capture spatial stream 1 carried on the 80+80 lower carrier.]
`:CONF:SRW:SEGM:APP CON80_80TMIMOM`	Create a capture segment for concurrent 80+80 T-MIMO measurements. This establishes the module as the "M" module for the measurement. This module will capture the lower 80+80 frequency segment of space-time stream 1.
`:CONF:SRW:SEGM:ACT 1`	Make segment 1 the active segment. ***Note:*** Strictly this is not necessary because the most recently added segment always becomes the active segment. However it is good practice to set the active segment explicitly when working with multiple segments.
`[:CONF:SRW:WLAC:SPEC:C80_80 COMBINED]`	Specify whether the 80+80 upper and lower frequency segments are combined in the DUT or output on separate antenna ports. ***Note 1:*** Use SEPARATE if the upper and lower 80+80 frequency segments are output from the DUT on separate antenna ports. If the signals are combined either in the DUT or externally using a combiner, set the value of this setting to COMBINED. The default value of this setting is COMBINED, so for this example the command is optional.
`:CONF:SRW:SEGM:PORT PORT3`	Assign the RF input port for the "M" module's measurements. It will also become the port assigned by default on the "S" modules when the CON80_80TMIMOS segments are created. In this example we are using port 3.
`:CONF:SRW:FREQ 5.29E+09`	Set the frequency of the lower 80+80 frequency segment to 5290 MHz.
`:CONF:SRW:CAPT:MODE PACKET`	Set the capture mode to Packet Count.

Command	Description
:CONF:SRW:PACK 10	Set the number of packets to be measured.
:CONF:SRW:TRIG IMMEDIATE	For this example the DUT will be set up to generate packets continuously. For that reason it is simpler to let the system locate and capture packets automatically by selecting IMMEDIATE triggering.
	[Segment 2 will capture the 80+80 upper frequency segment of space-time stream 1 concurrently using a separate module]
:CONF:SRW:SEGM:APP CON80_80TMIMOS	Create a capture segment for concurrent 80+80 MIMO measurements. The “S” suffix means that this segment will control an “S” or “slave” MU887000A module to capture the other 80+80 frequency segment of space-time stream 1.
:CONF:SRW:SEGM:ACT 2	Make segment 2 the active segment.
:CONF:SRW:SEGM:PORT PORT3	Set the port.
:CONF:SRW:FREQ 5.61E+09	Set the frequency to 5610 MHz, the upper frequency of the 80+80 transmission.
:CONF:SRW:SEGM:TMIM:IP "192.168.1.2"	Set the IP address of the “S” module. (This command is the same as is used for True MIMO measurements)
	[Segment 3 will capture the 80+80 lower frequency segment of space-time stream 2 concurrently using a separate module]
:CONF:SRW:SEGM:APP CON80_80TMIMOS	Create the capture segment
:CONF:SRW:SEGM:ACT 3	Make segment 3 the active segment.
:CONF:SRW:SEGM:PORT PORT3	Set the port.
:CONF:SRW:FREQ 5.29E+09	Set the frequency to 5290 MHz.
:CONF:SRW:SEGM:TMIM:IP "192.168.1.3"	Set the IP address of the “S” module.
	[Segment 4 will capture the 80+80 upper frequency segment of space-time stream 2 concurrently using a separate module]
:CONF:SRW:SEGM:APP CON80_80TMIMOS	Create the capture segment
:CONF:SRW:SEGM:ACT 4	Make segment 4 the active segment.
:CONF:SRW:SEGM:PORT PORT3	Set the port.
:CONF:SRW:FREQ 5.61E+09	Set the frequency to 5610 MHz.
:CONF:SRW:SEGM:TMIM:IP "192.168.1.4"	Set the IP address of the “S” module.
	[Setting up measurements]
:CONF:SRW:SEGM:ACT 1	Make the first segment the active segment. Any measurement settings on this segment will be applied automatically to segments 2, 3 and 4.

<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td>:CONF:SRW:SEL:WLAN:EVM:COMB ON
:CONF:SRW:SEL:WLAN:SPEC:GRAP ON</td><td>CON80_80TMIMOM and CON80_80TMIMOS segments have the following measurements enabled by default:

• Power
• EVM (Basic)
• Spectrum (Numeric)

Use :CONF:SRW:SEL... commands to enable and disable alternative sets of measurements. In this example we will enable combined EVM measurements and graphical spectrum measurements.

Note:
Any measurement enabled on the master segment is automatically enabled on the “S” segments.

So in this example you need only enable measurements on segment 1.

It is not necessary to enable “full span” spectrum measurements for 80+80 signals. This feature is enabled automatically.</td></tr>
<tr><td></td><td>[Autolevel]</td></tr>
<tr><td>:INIT:SRW:ALEV</td><td>Using the DUT manufacturer’s control software, set up the DUT to transmit 80+80 packets continuously, then:

Initiate Autolevel</td></tr>
<tr><td>*WAI</td><td>Wait for the operation to complete</td></tr>
</table>

6

Measurement Examples

Command	Description

`:CONF:SRW:POW?`
`:CONF:SRW:SEGM:ACT 2`
`:CONF:SRW:POW?`
`:CONF:SRW:SEGM:ACT 3`
`:CONF:SRW:POW?`
`:CONF:SRW:SEGM:ACT 4`
`:CONF:SRW:POW?`
 	
`:CONF:SRW:SEGM:ACT 1 <- See note.`
	This sequence of commands reads back the power set by the autolevel function on each module.
	
	You can use this response to confirm that the system is working as expected. For example, if the DUT typically generates a signal at +17 dBm, but the autolevel function returns –60 dBm, the reason could be that:
	
	• the wrong port has been selected for measurement on one of the MU887000A modules; or
	• the DUT has not been set up correctly; or
	• the DUT is not connected to the port selected for measurement.
	
	Note:
	You must set the active segment back to 1 before initiating the capture.

	[Capture]
`:INIT:SRW`	Initiate the capture.
`*WAI`	Wait for the operation to complete.
`:STAT:SRW:MEAS?`	Read the measurement status. It should be **1**, which indicates measurement complete.
	[Check Capture Information]
Refer to Section 7.1	Check that no errors occurred during the measurement.
	[Retrieve Measurements]

`:FETC:SRW:SUMM:WLAN:OFDM:EVM? 1,0`
`:FETC:SRW:SUMM:WLAN:OFDM:EVM? 2,0`
 	
`:FETC:SRW:SUMM:WLAN:OFDM:EVM:COMB? 1,0`
	Read the EVM measurement summary results for each segment. (The first command parameter is the segment number.)
	In this case the result from segment 1 will be the EVM of the lower frequency segment of space-time stream 1 and that from segment 2 will be the EVM of the upper frequency segment of space-time stream 1
	
	The combined EVM result can be fetched from either segment. In this example it is fetched from segment 1.

6.11.4 802.11ac Sequential 80+80 TMIMO Example – Two Modules

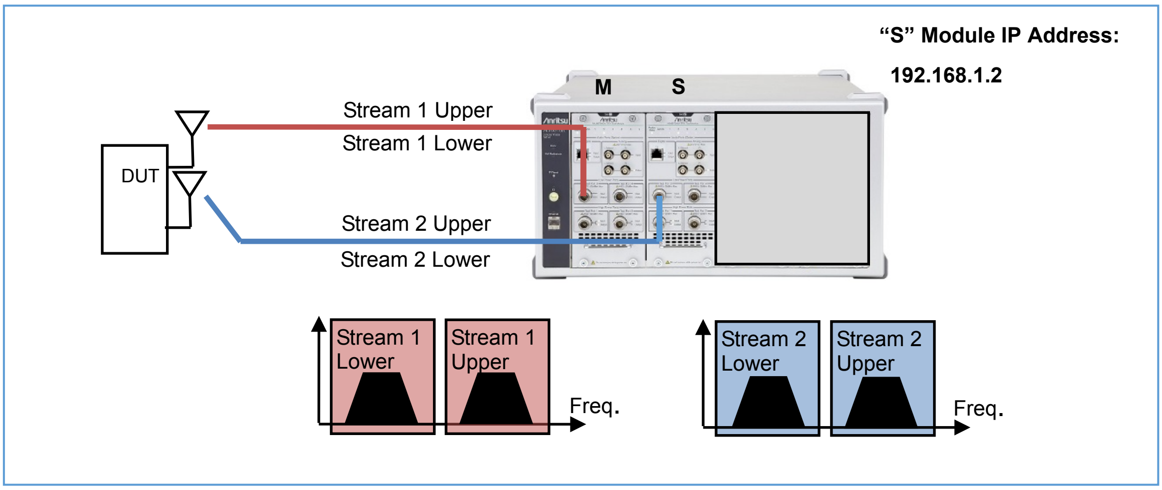


Figure 6.11.4-1 – 2x2 MIMO – 80+80 Signals Combined in DUT

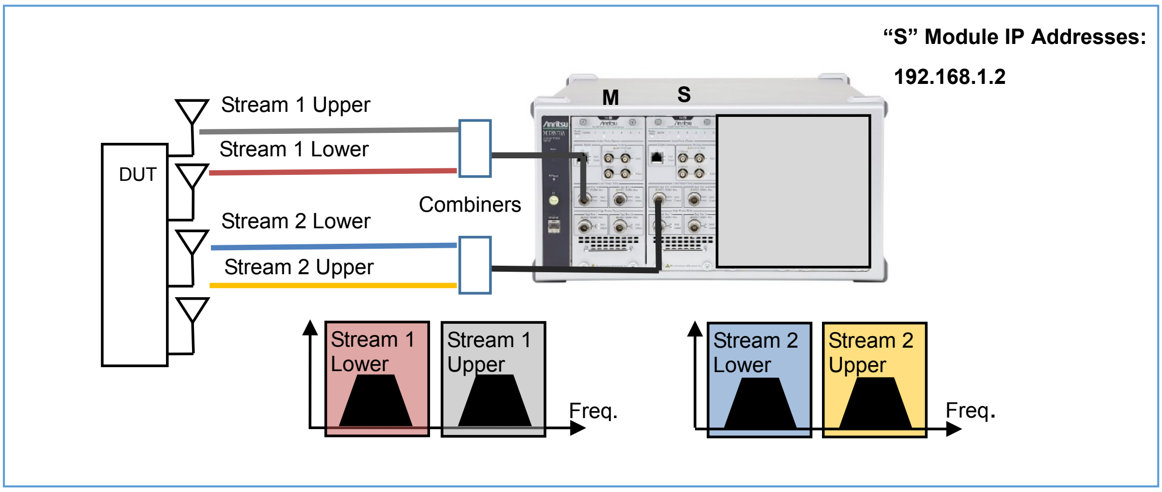


Figure 6.11.4-2 – 2x2 MIMO – 80+80 Signals on Separate DUT Ports

"Figure 6.11.4-1 – 2x2 MIMO – 80+80 Signals Combined in DUT"" shows the configuration for measuring a 2x2 MIMO 80+80 transmission using two MU87000A modules. There are two antenna ports, each carrying one of the two space-time streams. Each space-time stream is carried by an 80+80 signal.

"Figure 6.11.4-2 – 2x2 MIMO – 80+80 Signals on Separate DUT Ports" shows the set-up when the lower and upper 80+80 frequency segments forming each space-time stream are transmitted from separate antenna ports.

This example uses sequential 80+80 T-MIMO, which means that the lower and upper frequencies of the 80+80 signal are captured sequentially, but both of the space-time streams on each frequency are captured concurrently.

One MU887000A, the "master" module, designated "M", is in overall charge of the measurement. All measurement set-up commands are sent to this module and it returns all the measurement results. The other module, designated "S" is managed automatically by the "M" module.

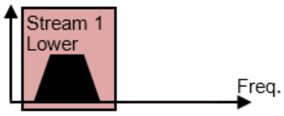
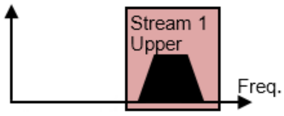
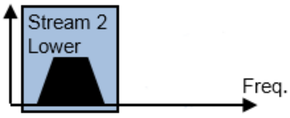
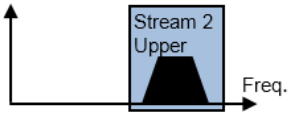
Notes:

- The "M" module must be located in the slot 1 or slot 3 position of the MT8870A mainframe.
- If the "M" module is in slot 1, the "S" module can be in any of the other slots of the same mainframe, or in a separate MT8870A mainframe provided that the second MT8870A is connected to the same local area network (LAN) and the **Trigger Input / Output 1** connectors on the rear panels of the mainframes are connected together.
- If the "M" module is in slot 3, the "S" module must be in slot 4 of the same mainframe.
- You need to know the IP addresses of the "S" module so that the "M" module can be set up to control it. In this example the IP addresses used is 192.168.1.2. Substitute the correct IP addresses for your "S" module.

The following example shows how to capture and measure an 80+80 2x2 MIMO signal with the following characteristics:

- Lower frequency segment: 5290 MHz
- Upper frequency segment: 5610 MHz
- Frequency segments are combined within the DUT. (See Figure 6.11.4-1 for connection diagram)
- Power level will be set up using Autolevel

Relationship between capture segments, measurements and time:

	Time →	
	First Capture	**Second Capture**
Segment 1	Measure space-time stream 1 on lower carrier: 5290 MHz on "M" module 	
Segment 2		Measure space-time stream 1 on upper carrier: 5610 MHz on "M" module 
Segment 3	Measure space-time stream 2 on lower carrier: 5290 MHz on "S" module 	
Segment 4		Measure space-time stream 2 on upper carrier: 5610 MHz on third "S" module 

Space-time streams 1 and 2 are captured concurrently, first at the lower carrier frequency, then at the upper carrier frequency.

All the following commands are sent to the MT887000A module designated as the "M" module:

Command	Description
	[Preliminaries when controlling a MU88700x]
`SYST:LANG SCPI`	Switch to SCPI mode ***Note:*** Do not use a colon character, ":", before this command.
`:INST SRW`	Select the SRW Application (WLAN / Bluetooth)
`:CALC:CAL:BAND:STAR:TEMP 2.0`	Run a calibration. In this case the calibration will only run if the temperature inside the instrument has changed by 2.0 degrees since the previous calibration.
`*WAI`	Wait for the calibration to complete. It takes few minutes to finish the calibration.
`:CALC:CAL:BAND:RES?`	Check the calibration was successful. The response should be **PASS**.
	[Set up four sequential 80+80 T-MIMO capture segments: **2 80+80 carriers x 2 space-time streams.]**
`:CONF:SRW:SEGM:CLE`	Clear any existing capture segments.
	[Segment 1 will capture the 80+80 lower frequency transmission]
`:CONF:SRW:SEGM:APP SEQ80_80TMIMOM`	Create a capture segment for sequential 80+80 T-MIMO measurements. This establishes the module as the "M" module for the measurement. This module will capture the lower 80+80 frequency segment of space-time stream 1.
`:CONF:SRW:SEGM:ACT 1`	Make segment 1 the active segment. ***Note:*** Strictly this is not necessary because the most recently added segment always becomes the active segment. However it is good practice to set the active segment explicitly when working with multiple segments.

Command	Description
[:CONF:SRW:WLAC:SPEC:C80_80 COMBINED]	Specify whether the 80+80 upper and lower frequency segments are combined in the DUT or output on separate antenna ports. ***Note 1:*** Use SEPARATE if the upper and lower 80+80 frequency segments are output from the DUT on separate antenna ports. If the signals are combined either in the DUT or externally using a combiner, set the value of this setting to COMBINED. The default value of this setting is COMBINED, so for this example the command is optional.
:CONF:SRW:SEGM:PORT PORT3	Assign the RF input port for the "M" module's measurements. It will also become the port assigned by default on the "S" module when its matching SEQ80_80TMIMOS segment is created. In this example we are using port 3.
:CONF:SRW:FREQ 5.29E+09	Set the frequency of the lower 80+80 frequency segment to 5290 MHz.
:CONF:SRW:CAPT:MODE PACKET	Set the capture mode to Packet Count.
:CONF:SRW:PACK 10	Set the number of packets to be measured.
:CONF:SRW:TRIG IMMEDIATE	For this example the DUT will be set up to generate packets continuously. For that reason it is simpler to let the system locate and capture packets automatically by selecting IMMEDIATE triggering.
	[Segment 2 is a second master segment that will capture the 80+80 upper frequency segment of space-time stream 1 and trigger a synchronous capture of space-time stream 2 on this frequency using the second module]
:CONF:SRW:SEGM:APP SEQ80_80TMIMOM	Create a capture segment for sequential 80+80 T-MIMO measurements. This segment will capture the upper 80+80 frequency segment of space-time stream 1.
:CONF:SRW:SEGM:ACT 2	Make segment 2 the active segment.
:CONF:SRW:SEGM:PORT PORT3	Set the port.
:CONF:SRW:FREQ 5.61E+09	Set the frequency to 5610 MHz, the upper frequency of the 80+80 transmission.

Command	Description
`:CONF:SRW:SEGM:TMIM:IP "192.168.1.2"`	Set the IP address of the "S" module. (This command is the same as is used for True MIMO measurements)
	[Segment 3 will capture the 80+80 lower frequency segment of space-time stream 2 concurrently using a separate module]
`:CONF:SRW:SEGM:APP SEQ80_80TMIMOS`	Create an "S" segment. This will execute in parallel with master segment 1 and will capture space-time stream 2 at the lower 80+80 frequency.
`:CONF:SRW:SEGM:ACT 3`	Make segment 3 the active segment.
`:CONF:SRW:SEGM:PORT PORT3`	Set the port.
`:CONF:SRW:FREQ 5.29E+09`	Set the frequency to 5290 MHz.
`:CONF:SRW:SEGM:TMIM:IP "192.168.1.2"`	Set the IP address of the "S" module.
	[Segment 4 will capture the 80+80 upper frequency segment of space-time stream 2 concurrently using a separate module]
`:CONF:SRW:SEGM:APP CON80_80TMIMOS`	Create an "S" segment. This will execute in parallel with master segment 2 and will capture space-time stream 2 at the upper 80+80 frequency.
`:CONF:SRW:SEGM:ACT 4`	Make segment 4 the active segment.
`:CONF:SRW:SEGM:PORT PORT3`	Set the port.
`:CONF:SRW:FREQ 5.61E+09`	Set the frequency to 5610 MHz.
`:CONF:SRW:SEGM:TMIM:IP "192.168.1.2"`	Set the IP address of the "S" module.
	[Setting up measurements]
`:CONF:SRW:SEGM:ACT 1`	Make the first segment the active segment. Any measurement settings on this segment will be applied automatically to segments 2, 3 and 4.

Command	Description
:CONF:SRW:SEL:WLAN:EVM:COMB ON :CONF:SRW:SEL:WLAN:SPEC:GRAP ON	SEQ80_80TMIMOM and SEQ80_80TMIMOS segments have the following measurements enabled by default: Power EVM (Basic) Spectrum (Numeric) Use :CONF:SRW:SEL… commands to enable and disable alternative sets of measurements. In this example we will enable combined EVM measurements and graphical spectrum measurements. ***Note:*** Any measurement enabled on the first master segment is automatically enabled on the second master segment and both the “S” segments. So in this example you need only enable measurements on segment 1. It is not necessary to enable “full span” spectrum measurements for 80+80 signals. This feature is enabled automatically.
	[Autolevel]
:INIT:SRW:ALEV	Using the DUT manufacturer’s control software, set up the DUT to transmit 80+80 packets continuously, then: Initiate Autolevel
*WAI	Wait for the operation to complete

6

Measurement Examples

Command	Description
:CONF:SRW:POW? :CONF:SRW:SEGM:ACT 2 :CONF:SRW:POW? :CONF:SRW:SEGM:ACT 3 :CONF:SRW:POW? :CONF:SRW:SEGM:ACT 4 :CONF:SRW:POW? :CONF:SRW:SEGM:ACT 1 <- See note.	This sequence of commands reads back the power set by the autolevel function on each module. You can use this response to confirm that the system is working as expected. For example, if the DUT typically generates a signal at +17 dBm, but the autolevel function returns –60 dBm, the reason could be that: • the wrong port has been selected for measurement on one of the MU887000A modules; or • the DUT has not been set up correctly; or • the DUT is not connected to the port selected for measurement. ***Note:*** You must set the active segment back to 1 before initiating the capture.
	[Capture]
:INIT:SRW	Initiate the capture.
*WAI	Wait for the operation to complete.
:STAT:SRW:MEAS?	Read the measurement status. It should be **1**, which indicates measurement complete.
	[Check Capture Information]
Refer to Section 7.1	Check that no errors occurred during the measurement.
	[Retrieve Measurements]
:FETC:SRW:SUMM:WLAN:OFDM:EVM? 1,0 :FETC:SRW:SUMM:WLAN:OFDM:EVM? 2,0 :FETC:SRW:SUMM:WLAN:OFDM:EVM:COMB? 1,0	Read the EVM measurement summary results for each segment. (The first command parameter is the segment number.) In this case the result from segment 1 will be the EVM of the lower frequency segment of space-time stream 1 and that from segment 2 will be the EVM of the upper frequency segment of space-time stream 1 The combined EVM result can be fetched from either segment. In this example it is fetched from segment 1.

6.12 Multi-Segment Transmitter Measurements

Some DUTs are capable of generating sequences of different waveforms for test purposes. This is often known as “List Mode”.

The SRW Application is able to analyze and return measurements of these sequences by using multi-segment captures. Each segment returns results for one type of waveform in the list generated by the DUT, as shown in Figure 6.12-1.

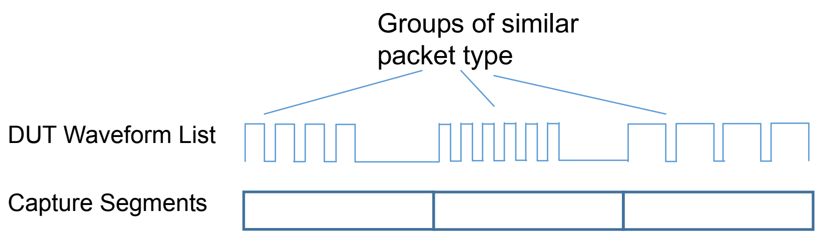


Figure 6.12-1 - Capturing a Waveform List using a Multi-Segment Capture

Questions and Answers

Q How do I synchronize the DUT and instrument?

A Set up level triggering on the first capture segment. This will ensure that the measuring system does not start capturing before the first packet in the DUT waveform list is received.

Q How should I set up each segment to match the DUT waveform list?

A Each group of packets in the DUT waveform list will map to a capture segment definition in the SRW Application. For example, if the DUT waveform list consists of a group of five 802.11g 54 Mbps packets followed by a group of five 802.11n MCS 7 packets, two segments must be defined - one for each group.

It is generally easier to use “Auto” segments (AUTOOFDM, AUTODSSS and AUTOBT). This minimizes the amount of set-up required.

Q Should I use packet counting or timed capture?

A Set the capture mode to “PACKET” in each segment. This allows the SRW application to count the packets generated by the DUT and switch to the next segment when the expected number has been received.

Q Is a settling delay required between segments?

A A settling delay of 500 μs is required at the start of any segment where any one of the following is true:

- The VSA frequency has changed.
- The power level setting has changed.
- The port has changed.

Q If a settling delay is required, how can I avoid missing packets from the DUT?

A Set up the DUT to leave a gap of at least 500 µs between groups of packets. Use packet count mode.

Q How can I be sure that the power level setting is correct in each segment?

A If the power levels of all groups within the DUT waveform list are within 5 dB of each other, it is possible to use the auto-level function. If there is a wider range of power levels, the levels must be set individually for each segment.

Q The DUT produces a number of unstable packets after changing frequency. How can these be excluded from the measurement?

A Use the packet skip feature to disregard a specified number of packets from the start of each segment.

To use this feature:

1. Set the number of “unstable” packets generated by the DUT.
2. Set the “packets-to-skip” value to this number.
3. Set the number of packets-to-measure.
4. Add the number of packets-to-measure to the number of packets-to-skip and program the DUT to generate a group of that number of packets within the waveform list.

Example

In this example we will assume that the DUT has been programmed to generate the following waveform list:

802.11g, 54 Mbps	5 packets
802.11n, 40 MHz, MCS7	5 packets
802.11ac, 80 MHz, MCS9	5 packets

We will also assume that there is little difference in power level between the three packet types, but that when the wireless standard is changed, the first packet produced does not meet specification and should be ignored.

To capture and measure this waveform list, set up the instrument as follows:

Command	Description
	[Preliminaries when controlling a MU88700x]
`SYST:LANG SCPI`	Switch to SCPI mode ***Note:*** Do not use a colon character, “:”, before this command.
`:INST SRW`	Select the SRW Application (WLAN / Bluetooth)
`:CALC:CAL:BAND:STAR:TEMP 2.0`	Run a calibration. In this case the calibration will only run if the temperature within the instrument has changed by 2.0 degrees since the previous calibration.
`*WAI`	Wait for the calibration to complete. It takes few minutes to finish the calibration.
`:CALC:CAL:BAND:RES?`	Check the calibration was successful. The response should be **PASS**.
	[Set up a multi-segment capture]
`:CONF:SRW:SEGM:CLE`	Clear any existing capture segments.
	[Segment 1 - will capture the 802.11g packets]
`:CONF:SRW:SEGM:APP AUTOOFDM`	Create a capture segment for any WLAN OFDM packet.
`:CONF:SRW:SEGM:ACT 1`	Make segment 1 the active segment. ***Note:*** Strictly this is not necessary because the most recently added segment always becomes the active segment. However it is good practice to set the active segment explicitly when working with multiple segments.
`:CONF:SRW:PORT PORT3`	Set the port.
`:CONF:SRW:FREQ 5E+09`	Set the frequency to 5 GHz.
`:CONF:SRW:POW <Expected power>`	[Optional] Set the power level expected from the DUT. ***Note:*** Autolevel should work well provided that the DUT power level variation throughout the list does not exceed 5 dB. If the power level is set explicitly and varies from segment to segment, a 500 μs settling delay is required at the start of each segment. You should allow a gap of at least 500 μs between the packet groups generated by the DUT to avoid missing packets.
`:CONF:SRW:CAPT:MODE PACKET`	Set the capture mode to Packet Count.
`:CONF:SRW:PACK 4`	Set the number of packets to be measured.
`:CONF:SRW:PACK:SKIP 1`	Set the number of packets to be skipped.

Command	Description
`:CONF:SRW:TRIG LEVEL`	Use level triggering on the first segment so that the DUT list and instrument capture start together.
	[Segment 2 - will capture the 802.11n packets]
`:CONF:SRW:APPEND AUTOOFDM`	Create a capture segment for any WLAN OFDM packet.
`:CONF:SRW:SEGM:ACT 2`	Make segment 2 the active segment.
`:CONF:SRW:PORT PORT3`	Set the port.
`:CONF:SRW:FREQ 5E+09`	Set the frequency to 5 GHz.
`:CONF:SRW:POW <Expected power>`	See notes for Segment 1.
`:CONF:SRW:CAPT:MODE PACKET`	Set the capture mode to Packet Count.
`:CONF:SRW:PACK 4`	Set the number of packets to be measured.
`:CONF:SRW:PACK:SKIP 1`	Set the number of packets to be skipped.
`:CONF:SRW:TRIG IMMEDIATE`	Use immediate triggering ***Note:*** Because we are using Packet Count mode, level triggering is not required for segment 2 or later.
	[Segment 3 - will capture the 802.11ac packets]
`:CONF:SRW:APPEND AUTOOFDM`	Create a capture segment for any WLAN OFDM packet.
`:CONF:SRW:SEGM:ACT 3`	Make segment 3 the active segment.
`:CONF:SRW:PORT PORT3`	Set the port.
`:CONF:SRW:FREQ 5E+09`	Set the frequency to 5 GHz.
`:CONF:SRW:POW <Expected power>`	See notes for Segment 1.
`:CONF:SRW:CAPT:MODE PACKET`	Set the capture mode to Packet Count.
`:CONF:SRW:PACK 4`	Set the number of packets to be measured.
`:CONF:SRW:PACK:SKIP 1`	Set the number of packets to be skipped.
`:CONF:SRW:TRIG IMMEDIATE`	Use immediate triggering.
	[Autolevel]
`:INIT:SRW:ALEV`	Initiate Autolevel ***Note:*** Use this command if the power level of each type of packet transmitted by the DUT lies within a 5 dB range. Otherwise set the level for each segment independently. The DUT must be set to transmit its list in a loop before initiating autolevel
`*WAI`	Wait for the operation to complete
`:CONF:SRW:POW?`	Read back the power level set by the autolevel function. All segments will be set to the same level.

Command	Description
	[Capture]
`:INIT:SRW`	Initiate the capture. ***Note:*** There are three steps: 1. Ensure that the DUT is not transmitting. 2. Send the command. The instrument should be waiting for an RF trigger. 3. Set the DUT to transmit its list.
`*WAI`	Wait for the operation to complete.
`:STAT:SRW:MEAS?`	Read the measurement status. It should be **1**, which indicates measurement complete.
	[Check Capture Information]
Refer to Section 7.1	Check that no errors occurred during the measurement.
	[Retrieve Measurements]

`:FETC:SRW:SUMM:WLAN:OFDM:EVM? 1,0`
`:FETC:SRW:SUMM:WLAN:OFDM:EVM? 2,0`
`:FETC:SRW:SUMM:WLAN:OFDM:EVM? 3,0`
	Read the EVM measurement summary results for each of the three segments. (The first command parameter is the segment number.)
	Note:
	AUTOOFDM segments have the following measurements enabled by default:
	• Power
	• EVM (Basic)
	• Spectrum (Numeric)
	
	Use :CONF:SRW:SEL… commands to enable and disable alternative sets of measurements. Measurements must be configured separately for each segment.

Chapter 7 Error Handling

This chapter explains how to check for errors using the capture information and status data.

7.1 Checking for Errors

7.1.1 Using Capture Information

Following a capture, the `FETCh:SRWireless:CINFormation?` command should be used to determine whether the capture was successful and to find the cause of any problems encountered.

If there were any problems during the capture, bit 9 is set in the Questionable Condition register. Details of the Questionable register can be found in section 3.6.6 of the *MU887000A TRX Test Module Operation Manual*.

The response to a `FETCh:SRWireless:CINFormation?` request is described in the table below.

Field	Description
Capture status	Overall capture status
Number of segments	Number of segments captured
Segment index of first segment	Index number of the first segment in the capture, starting from 1
Status of first segment	Status of the segment
Offset from start of capture	Location of the segment in IQ pairs from the start of the capture.
Width of segment	Width of segment in IQ pairs
Number of packets in segment	Actual number of packets found within the segment.
Packet index of first packet in segment	Index number of the first packet in the segment, counting from 1.
Status of first packet	Status of the packet
Offset of packet from start of segment	Location of the packet in IQ pairs from the start of the segment.
Width of packet	Width of the packet in IQ pairs.
Packet index of second packet in segment	Index number of the second packet in the segment.
…	Information on second and subsequent packets within the first segment as above.
Segment index of 2nd segment	Index number of the second segment in the capture.
…	Information on the second and subsequent segments (and their packets) as above.

Note:

An "IQ Pair" is a 32-bit I value and a 32-bit Q value. An IQ pair occupies 8 bytes of memory.

The status information can be used as shown in the table below.

Field	Value	Meaning
Capture Status	0	No problems were found anywhere within the capture.
	1	At least one problem was found in a segment or a packet within a segment. Examine the segment status fields for more information.
	> 1	The number relates to a specific problem with the capture. See 7.1.2 Error Codes.
Segment Status	0	No problems were found anywhere within this segment.
	1	At least one problem was found in a packet within this segment. Examine the packet status fields for more information.
	> 1	The number relates to a specific problem associated with segments. See 7.1.2 Error Codes.
Packet Status	0	No problems were found in this packet.
	> 0	The number relates to a specific problem associated with the packet. See 7.1.2 Error Codes.

7.1.2 Error Codes

General Errors: 0x0001 - 0x00FF

Value	Summary
0 (0x0000)	No errors
2 (0x0002)	Memory allocation error
3 (0x0003)	Unrecognized segment type
4 (0x0004)	Sampling frequency not supported
5 (0x0005)	Invalid settings
6 (0x0006)	Invalid packet
7 (0x0007)	Invalid wireless standard
8 (0x0008)	Invalid gate settings
9 (0x0009)	Invalid gate offset
10 (0x000a)	Invalid results type
11 (0x000b)	Invalid configuration
12 (0x000c)	No results available
13 (0x000d)	No segments found
14 (0x000e)	Segment configuration failure
15 (0x000f)	Capture size too long
16 (0x0010)	No packets found
17 (0x0011)	Burst power not found
18 (0x0012)	Internal segment error
19 (0x0013)	Unknown packet
20 (0x0014)	Incompatible packet
21 (0x0015)	A composite MIMO measurement cannot be initiated because reference data has not been loaded.
22 (0x0016)	The composite MIMO reference data format is not supported. The most likely reason for this error is that the reference data format is incompatible with the SRW Application firmware installed on the instrument. A firmware upgrade may be required.
23 (0x0017)	Composite MIMO reference data can only be used with composite MIMO (COMPMIMO) segments. It is invalid to attempt to load reference data for any other type of segment.
24 (0x0018)	Invalid number of segments. Some operations require a specific number of segments to be configured. For example, a true MIMO measurement requires one "M" segment and between 1 and 7 "S" segments. 802.11ac 80+80 measurements require segments to be defined in pairs to capture the primary and secondary frequency segments.
25 (0x0019)	No license. A license is required to complete the operation. For example, an 802.11ac measurement requires the relevant license to be installed in the MT8870A mainframe.

Capture Errors: 0x0200 - 0x02FF

Value	Summary
512 (0x200)	Empty capture (Note: not used)
513 (0x201)	Capture configuration error
514 (0x202)	Error stop notify (hardware problem).
515 (0x203)	Power range error. (Over-range condition detected. Note LED 3 turns red and instrument beeps.)
516 (0x204)	Leveling error. (Reserved for future use. No error reports are necessary with current leveling implementation.)
517 (0x205)	Trigger timeout. (Note: LED 3 turns red and instrument beeps.)
518 (0x206)	Reserved
519 (0x207)	Out of capture memory. There is insufficient capture memory to support the current segment definitions.
520 (0x208)	Capture data transfer error. An internal error was detected in the capture system. If it occurs repeatedly it may a hardware problem. Contact Anritsu Technical Support for advice.
521 (0x209)	Capture data memory full. This error can only occur when the capture consists of more than one capture segment and packet counting is used. The error occurs if the segment capture starts, but insufficient packets are counted before the capture memory is full, thus preventing the capture from completing. Check that the packet count value matches the number of packets expected from the DUT.
522 (0x20A)	Capture data overflow. An internal error was detected in the capture system. If it occurs repeatedly it may a hardware problem. Contact Anritsu technical support for advice.

Power Measurement Errors

Value	Summary
768 (0x0300)	Packet of zero length. (Packet too short for power measurement.)

Frequency Measurement Errors

Value	Summary
1024 (0x0400)	Insufficient data (The minimum capture time for a frequency measurement is 80 μs)

Spectrum Measurement Errors

Value	Summary	Detail
1280 (0x0500)	Insufficient data.	A spectrum measurement requires a sample data length of at least 50 µs
1281 (0x0501)	Unknown spectral profile	Applies to WLAN OFDM and DSSS auto-ID modes. The system failed to identify a signal of 20 MHz, 40 MHz, 80 MHz or 160 MHz.
1282 (0x0502)	Full span error	Full span spectrum profiles covering a span of ±140 MHz are produced by making three separate spectrum measurements: one at the carrier frequency, one at an offset of 60 MHz below the carrier and one at an offset of 60 MHz above. This error is reported if the full span measurement fails. A likely cause is that the instrument receiver is not accurately tuned to the DUT transmitter, and one of the three spectrum measurements has failed to detect WLAN packets.
1283 (0x0503)	Frequency offset outside range	A spot frequency reading was attempted with the offset from the carrier exceeding the frequency span of the measurement.

OFDM Measurement Errors

Value	Summary	Detail
1536 (0x0600)	Sync error. Could not sync to OFDM packet.	
1537 (0x0601)	"Invalid setting. (For example, a short guard interval is not allowed with Greenfield.)"	
1538 (0x0602)	Bandwidth detector failed	The system failed to recognize the OFDM signal bandwidth as 20 MHz, 40 MHz, 80 MHz or 160 MHz
1539 (0x0603)	Packet header detection failed	
1540 (0x0604)	L-SIG parity check failed	
1541 (0x0605)	Unknown data rate	
1542 (0x0606)	HT-SIG CRC check failed	
1543 (0x0607)	VHT-SIG CRC check failed	
1544 (0x0608)	Unknown VHT bandwidth (Not used)	
1545 (0x0609)	Unknown HT bandwidth	The packet header indicates that it is an HT packet, but the measured signal bandwidth is other than 20 MHz or 40 MHz
1546 (0x060A)	Unsupported 802.11n packet	The MCS index of the packet is not supported.

Value	Summary	Detail
1547 (0x060B)	Unsupported 802.11ac packet	The number of MIMO spatial streams is greater than four.
1548 (0x060C)	Invalid 802.11n packet	
1549 (0x060D)	Invalid 802.11ac packet	
1550 (0x060E)	Unsupported multi-user 802.11ac packet	The packet header indicates that it is a multi-user packet, which is not supported
1551 (0x060F)	Composite MIMO unsupported standard	
1552 (0x0610)	Composite MIMO unsupported number of streams	
1553 (0x0611)	Composite MIMO constellation length error	
1554 (0x0612)	Composite MIMO - received packet and reference file mismatch	The received packet did not match the reference file. The DUT may have been set up incorrectly or the wrong reference file selected.
1555 (0x613)	MIMO: incompatible number of spatial streams	The number of spatial streams expected did not match the number decoded from the packet header.
1556 (0x614)	MIMO: not a MIMO packet	The received packet was not a MIMO packet.
1557 (0x615)	MIMO: incompatible streams	The detected streams were incompatible. For example, the MCS index may differ between streams. (This can happen by mistake when a MIMO device is simulated using signal generators.)
1558 (0x616)	MIMO: inconsistent bandwidth	Not all spatial streams have the same bandwidth.
1559 (0x617)	MIMO: duplicate streams	The same spatial stream has been detected on more than one input port.
1560 (0x618)	Insufficient symbols in packet to measure symbol clock error	There must be at least two symbols in a packet to measure the symbol clock error.
1561 (0x619)	MIMO - failure detected on one or more streams.	Enhanced EVM measurements are made by simultaneously processing packets obtained from each MIMO stream. This error is reported if the system is unable to process packets from one or more of the streams.
1569 (0x621)	Unsupported 11ax packet	The packet header indicates that it is 11ax signal. The selected segment does not support 11ax measurement.
1570 (0x622)	Unsupported 11ax SU PPDU packet	The packet header indicates that it is 11ax SU PPDU signal, which is not supported.
1571 (0x623)	Unsupported 11ax MU PPDU packet	The packet header indicates that it is 11ax MU PPDU signal, which is not supported.
1572 (0x624)	Unsupported 11ax ERSU PPDU packet	The packet header indicates that it is 11ax ERSU PPDU signal, which is not supported.
1573 (0x625)	Unsupported 11ax TB PPDU packet	The packet header indicates that it is 11ax TB PPDU signal, which is not supported.

DSSS Measurement Errors: 0x0700 - 0x07FF

Value	Summary	Detail
1792 (0x0700)	Failed to correlate packet header	
1793 (0x0701)	Could not synchronize to the Start of Frame delimiter	
1794 (0x0702)	User EVM analysis length error. (User setting exceeds the actual length available.)	
1795 (0x0703)	Packet CRC failure	
1796 (0x0704)	Chip clock analysis length error. (User setting exceeds the actual length available in the packet.)	
1797 (0x0705)	Packet length mismatch	The packet length decoded from the packet header does not match the length of packet as determined by the analysis system.
1798 (0x0706)	Invalid configuration setting	

Bluetooth Measurement Error: 0x0800 - 0x08FF

Value	Summary	Detail
2048 (0x0800)	Could not find P0 in the GFSK header	Failed to find P0 (beginning of packet header). This can indicate that the Bluetooth packet is badly constructed or that it is some other kind of packet, such as WLAN.
2049 (0x0801)	The packet type set by the user is incompatible with the requested measurement. For example it is not possible to return an eye diagram measurement result if it is an EDR packet type.	
2050 (0x0802)	Failed to find a power burst within the packet IQ data samples.	
2051 (0x0803)	GFSK header synchronization failure	Header synchronization occurred outside the expected bounds of the packet header. The packet may not be a Bluetooth packet or the packet's Bluetooth address may not match the expected address.
2052 (0x0804)	Failed to decode EDR packet type	
2053 (0x0805)	EDR header correlation error	Failed to find start of the EDR payload. This can indicate that the packet is not an EDR packet.
2054 (0x0806)	EDR payload length mismatch. (The payload length does not match the value set by the user)	

Value	Summary	Detail
2055 (0x0807)	EDR block length mismatch. Could not compute number of 50 µs blocks.	
2056 (0x0808)	Basic rate invalid packet type. (The packet type set by the user is not a basic rate type.)	
2057 (0x0809)	Basic rate payload length mismatch. (The received packet payload length does not match that set by the user.	
2058 (0x080a)	Basic rate carrier drift – invalid payload type. (SIG Standard mode only: to be compliant the payload type must be 10101010.)	
2059 (0x080b)	Basic rate modulation – invalid payload type. (SIG Standard mode only: to be compliant the payload type must be 10101010 or 11110000)	
2060 (0x080c)	BLE invalid packet type. (The packet type set in the measurement configuration is not a BLE type.)	
2061 (0x080d)	BLE payload length mismatch. (The received packet payload length does not match that set by the user.)	
2062 (0x080e)	BLE carrier offset and drift – invalid payload type. (SIG Standard mode only: to be compliant the payload type must be 10101010.)	
2063 (0x080f)	BLE modulation – invalid payload type. (SIG Standard mode only: to be compliant the payload type must be 10101010 or 11110000.)	
2064 (0x0810)	Insufficient samples. The IQ data length is insufficient for analysis. (This could be caused by a packet fragment in the IQ data.)	
2065 (0x0811)	Limit checking error. Could not perform limit checking because of numeric overflow.)	
2066 (0x0812)	Basic rate packet length mismatch.	The payload length field in the packet header does not match the actual length of the payload data.

CCDF Measurement Errors

Value	Summary	Detail
2304 (0x0900)	Insufficient gate width	The gate width must be 10 µs or greater.

Power Profile Errors

Value	Summary	Detail
2560 (0x0a00)	Insufficient data for power profile measurement	The packet length must be at least 1µs. Short noise spikes may be being misinterpreted as packets. Try raising the packet detector threshold by a few dB (for example from –20 dB to –17 dB).
2561 (0x0a01)	Insufficient data for power ramp measurement	The packet length must be at least 20 µs. This error may indicate that the packet detector is misinterpreting short noise bursts as a valid packet. Try raising the packet detector threshold by a few dB (for example from –20 dB to –17 dB).

IPP Errors (Intel Performance Primitives)

Value	Description
2816 (0x0b00)	IPP operation error. Please contact Anritsu Technical Support if this error persists.

True MIMO Errors

Value	Summary	Detail
3072 (0x0c00)	A maximum of 7 “S” modules are supported	An attempt has been made to connect to more than 7 external modules ("S" modules).
3073 (0x0c01)	Failed to connect	Failed to connect to an "S" module. A possible cause is that its IP address has not been entered correctly.
3074 (0x0c02)	Already connected	A connection to an "S" module has already been established.
3075 (0x0c03)	Wrong segment type	The requested operation cannot be performed on this type of segment.
3076 (0x0c04)	Communications error	An error was detected when programming an “S” module. (Examine the logs of the “M” and “S” modules to determine the error.)
3077 (0x0c05)	IP address error	The IP address is badly formed (e.g. 192.168.0.300) or it begins 0.x.x.x or 127.x.x.x
3078 (0x0c06)	Duplicate IP address	The IP address has already been used.
3079 (0x0c07)	Master module memory full.	There is not enough memory in the Master ("M") module to hold data captured by the Slave ("S") modules. In practice this means that the sum of the capture times on all the "S" modules used for the measurement exceeds the remaining capture time available in the "M" module.

General Warning-Level Codes

Value	Description
32768 (0x8000)	

A.1 MX887030A Specification

Item	Specification
Product Number	MX887030A
Product Name	WLAN 802.11b/g/a/n TX Measurement MX887030A
Use application	The MX887030A WLAN 802.11b/g/a/n TX Measurement option are designed for measuring the transmitter of IEEE802.11 b, g, a, n(1X1) HT 20/40MHz radios when this application is installed in MU887000A TRX Test module and using either port 3 or 4. IEEE802.11a, 802.11j ,802.11n 1x1 (5GHz) require the 6GHz option (MU887000A-001) Definition of terms. CAL: CAL is the self-calibration function in the measurement instrument. The specifications are applicable after 30 minutes warm up at constant ambient temperature unless otherwise specified.
Product configuration	MX887030A 1 DVD-R 1 Operation Manual 1

Item	Specification
Electrical characteristic	Specifications are for Port 3 and 4 unless otherwise stated.
Common Item	
Measuring Object	WLAN Signal Packet
Frequency Range	2.4GHz Band: 2412 to 2484MHz 5GHz Band: 4920 to 5825MHz (Required MU887000A-001 option)
RF Power	
Input Setting Range	–65 to +25 dBm (Port 3/4)
Measurement Accuracy	After calibration (at 20 to 30°C) ±0.7 dB (–30 dBm ≤ Level ≤ +25 dBm) ±1.0 dB (–50 dBm ≤ Level < –30 dBm)
Power measurement bandwidths	40/20 MHz 802.11n 20 MHz 802.11b/g/a
Capture time	1.34 s
Pre-trigger	1.33 s
Resolution (time domain profile)	5 ns/sample
CCDF	CCDF defined as a percentage of samples against dB, where percentage of samples is normalized to the average power in the 'gate', and dB is defined as the relative value of samples greater than the average.
Power distribution value	A single numeric value called the power distribution value defines the number of dB above the average power below which a user defined percentage of the total number of samples falls.

Item	Specification
Spectral profile measurements	
Spectral profile measurement span	±65 MHz 802.11n ±35 MHz 802.11b/g/a
Minimum capture time	50 µs
Input signal measurement range with set RBW (100 kHz)	–27 to +25 dBm
Linearity	CW,RBW = 100 kHz Same as MU887000A Level Linearity port 3/4. ±0.2 dB (0 to –40 dB, ≥ –55 dBm)
Resolution	0.1 dB
Measurement bandwidth	100 kHz
EVM (Modulation Accuracy)	
EVM Measurement Range	–20 to +25 dBm
Residual EVM	
DSSS	Signal Level > –20 dBm, averaged over 20 packets < –28 dB
OFDM	Signal Level > –20 dBm, averaged over 20 packets Channel Estimation: FULLPACKET < –40 dB
EVM Data Format	dB or %
Measurement resolution	0.1% or 0.1 dB. All limit checking in dB to 0.1 dB resolution
Measurement speed	> 20 readings/second

Item	Specification
DSSS EVM Measurement settings	
RX filter type	Selectable between none, Gaussian, Root Raised Cosine
Gaussian Filter setting BT	BT 0.3 to 1.0, default 0.5, resolution 0.1
Root Raised Cosine Filter setting	α 0.30 to 1.00, default 0.35, resolution 0.01
Measurement start	It shall be possible to measure EVM from the first data chip of the packet
Measurement method	Header or payload. Header measures the EVM of the first 1000 chips of the PLCP preamble and header.
User Specified Measurement Range	220 to 11000 chips, default 1000 chips
Measurement Functional Range	Measurement only possible if channel frequency error < ±150 kHz (±60 ppm)
Carrier Lock	Phase tracking automatically applied as per carrier lock 802.11–2007 18.4.7.8
OFDM EVM Measurement Settings	
Channel estimation	User selection of Long Training Sequence or Full Packet. Default Long Training sequence
User Specified measurement Range	Minimum 16 Max 1000 symbols, default 40
OFDM Pilot tracking	"Phase tracking only" or "Phase and Amplitude Tracking". Default Phase tracking only Peak and Average EVM on all sub-carriers, dB or percentage Peak and Average on each sub-carrier – frequency domain % v sub-carrier EVM v Symbol – time domain % v Symbol number, 1 to max

Item	Specification
DSSS Additional measurements	
Transmit center frequency tolerance	
Definition	Average frequency of the DSSS carrier signal
Accuracy	± (Setting frequency × Reference oscillator accuracy + 1 kHz)
Resolution	Hz to 1 Hz, ppm to 1 decimal place
Chip Clock Frequency Tolerance	
Definition	Frequency error relative to the 11 MHz Chip clock. Measurement averaged over a fully coded DSSS packet with minimum payload length 3300 chips, 300 μs
Display format	Hz and ppm
Range	±50 ppm
Resolution	Hz to 1 Hz, ppm to 1 decimal place
Data Analysis width	20 μs (220 chips) minimum
User Specified measurement range	3300 to 30250 chips, default 5500 chips
Transmit power-on and power down ramp	
Definition	Time for burst to transit from 10% to 90% or 90% to 10% of linear power.
Data outputs	10%, 90% and delta values
Resolution	5 ns
RF Carrier Suppression	
Method	Switchable between IEEE Std 802.11-2007 (18.4.7.7) or IQ offset method
IEEE method	Relative level of the carrier to the highest sideband for a 10101010 test pattern with scrambler disabled, data rate 2 Mbps.
IQ Offset method	Calculated from the relative values of the peak frequency response and the channel center frequency with the data rate processing gain.

Item	Specification
OFDM Additional Measurements	
Transmit center frequency tolerance	
Definition	Average frequency of the OFDM carrier signal
Data output format	Hz and ppm
Accuracy	> 1 ms packet ± (Setting frequency × Reference oscillator accuracy + 1 kHz)
Resolution	Hz to 1 Hz, ppm to 1 decimal place
Symbol clock frequency tolerance	
Definition	Frequency error relative to the 250 kHz symbol clock as per 19.4.7.3/17.3.9.5 Measurement averaged over a fully coded OFDM packet with a minimum payload length of 16 symbols (64 μs)
Data output format	Hz and ppm
Range	±40 ppm
Resolution	ppm to 1 decimal place
User specified measurement range	16 to (define numbers), default 55
Transmitter center frequency leakage	
Definition	Measurement of the leakage of the center carrier
Data output format	dB
Resolution	dB to two decimal places

Item	Specification
Transmitter spectral flatness	
Definition	Measurement of RF sub-carrier power level
Unit of measurement	dB
Additional Measurements (DSSS and OFDM)	
Power Spectral density	The maximum power measured in a 1 MHz bandwidth within the occupied bandwidth of the signal
Occupied bandwidth	Measures the frequency range within which the specified percentage power is contained
Occupied bandwidth percentage range	1 to 99%
Reference specification	
802.11b Measurements	
Transmit Power Levels	IEEE 802.11-2007 (18.4.7.1)
Transmit Power Level Control	IEEE 802.11-2007 (18.4.7.2)
Transmit Spectrum Mask	IEEE 802.11-2007 (18.4.7.3)
Transmit Center Frequency Tolerance	IEEE 802.11-2007 (18.4.7.4)
Chip Clock Frequency Tolerance	IEEE 802.11-2007 (18.4.7.5)
Transmit power-on and power-down ramp	IEEE 802.11-2007 (18.4.7.6)
RF Carrier Suppression	IEEE 802.11-2007 (18.4.7.7)
Transmit Modulation Accuracy	IEEE 802.11-2007 (18.4.7.8)

Item	Specification
802.11g/a/n Measurements	
Transmitter Power Levels	IEEE 802.11-2007 (17.3.9.1) IEEE 802.11-2007 (19.4.7.1) IEEE 802.11n-2009 (20.3.21.3)
Transmitter Spectrum Mask	IEEE 802.11-2007 (17.3.9.2) IEEE 802.11-2007 (19.5.4) IEEE 802.11n-2009 (20.3.21.1)
Transmit center frequency tolerance	IEEE 802.11-2007 (17.3.9.4) IEEE 802.11-2007 (19.4.7.2) IEEE 802.11n-2009 (20.3.21.4)
Symbol Clock frequency tolerance	IEEE 802.11-2007 (17.3.9.5) IEEE 802.11-2007 (19.4.7.3) IEEE 802.11n-2009 (20.3.21.6)
Transmitter center frequency leakage	IEEE 802.11-2007 (17.3.9.6.1) IEEE 802.11n-2009 (20.3.21.7.2)
Transmitter spectral flatness	IEEE 802.11-2007 (17.3.9.6.2) IEEE 802.11n-2009 (20.3.21.2)
Transmitter modulation accuracy	IEEE 802.11-2007 (17.3.9.6.3) (17.3.9.7) IEEE 802.11-2007 (19.7.2.7) IEEE 802.11n-2009 (20.3.21.7.3) (20.3.21.7.4)

A.2 MX887031A Specification

Item	Specification
Product Number	MX887031A
Product Name	WLAN 802.11ac TX Measurement MX887031A
Use application	The MX887031A WLAN 802.11ac TX Measurement option are designed for measuring the transmitter of IEEE802.11ac when this application is installed in MU887000A TRX Test module and using either port 3 or 4. IEEE802.11ac require the 6GHz option (MU887000A-001) Definition of terms. CAL: CAL is the self-calibration function in the measurement instrument. The specifications are applicable after 30 minutes warm up at constant ambient temperature unless otherwise specified.
Product configuration	MX887031A 1 DVD-R 1 Operation Manual 1

Item	Specification
Electrical characteristic	Specifications are for Port 3 and 4 unless otherwise stated.
Common Item	
Measuring Object	WLAN Signal Packet
Frequency Range	5GHz Band: 4920 to 5825 MHz (Required MU887000A-001 option)
RF Power	
Input Setting Range	–65 to +25 dBm (Port 3/4)
Measurement Accuracy	After calibration (at 20 to 30°C) ±0.7 dB (–30 dBm ≤ Level ≤ +25 dBm) ±1.0 dB (–50 dBm ≤ Level < –30 dBm)
Power measurement bandwidths	160 MHz / 80 MHz / 40 MHz / 20 MHz
Capture time	1.34 s
Pre-trigger	1.33 s
Resolution (time domain profile)	5 ns/sample
CCDF	CCDF defined as a percentage of samples against dB, where percentage of samples is normalized to the average power in the ‘gate’, and dB is defined as the relative value of samples greater than the average.
Power distribution value	A single numeric value called the power distribution value defines the number of dB above the average power below which a user defined percentage of the total number of samples falls.

Item	Specification
Spectral profile measurements	
Spectral profile measurement span	±80 MHz
Minimum capture time	50 µs
Input signal measurement range with set RBW (100 kHz)	–27 to +25 dBm
Linearity	CW,RBW = 100 kHz ±0.2dB(0 to –40dB, ≥ –55dBm)
Resolution	0.1 dB
Measurement bandwidth	100 kHz
EVM (Modulation Accuracy)	
EVM Measurement Range	–20 to +25 dBm
Residual EVM	
≤ 80 MHz	Signal Level > –10 dBm, averaged over 20 packets Channel Estimation: FULLPACKET < –38 dB
EVM Data Format	dB or %
Measurement resolution	0.1% or 0.1 dB. All limit checking in dB to 0.1 dB resolution
Measurement speed	> 20 readings/second

Item	Specification
OFDM EVM Measurement Settings	
Channel estimation	User selection of Long Training Sequence or Full Packet. Default Long Training sequence
User Specified measurement Range	Minimum 16 Max 1000 symbols, default 40
OFDM Pilot tracking	“Phase tracking only” or “Phase and Amplitude Tracking”. Default Phase tracking only Peak and Average EVM on all sub-carriers, dB or percentage Peak and Average on each sub-carrier – frequency domain % v sub-carrier EVM v Symbol – time domain % v Symbol number, 1 to max
OFDM Additional Measurements	
Transmit center frequency tolerance	
Definition	Average frequency of the OFDM carrier signal
Data output format	Hz and ppm
Accuracy	> 1 ms packet ± (Setting frequency × Reference oscillator accuracy + 1 kHz)
Resolution	Hz to 1 Hz, ppm to 1 decimal place
Symbol clock frequency tolerance	
Definition	Frequency error relative to the 250 kHz symbol clock as per 19.4.7.3/17.3.9.5 Measurement averaged over a fully coded OFDM packet with a minimum payload length of 16 symbols (64 μs)
Data output format	Hz and ppm
Range	±40 ppm
Resolution	ppm to 1 decimal place
User specified measurement range	16-to (define numbers), default 55

Item	Specification
Transmitter center frequency leakage	
Definition	Measurement of the leakage of the center carrier
Data output format	dB
Resolution	dB to two decimal places
Transmitter spectral flatness	
Definition	Measurement of RF sub-carrier power level
Unit of measurement	dB
Reference specification	
Transmit spectrum mask	IEEE P802.11ac/D2.0, January 2012 (22.3.18.1)
Spectral flatness	IEEE P802.11ac/D2.0, January 2012 (22.3.18.2)
Transmit center frequency tolerance	IEEE P802.11ac/D2.0, January 2012 (22.3.18.3)
Symbol clock frequency tolerance	IEEE P802.11ac/D2.0, January 2012 (22.3.18.4)
Modulation accuracy	IEEE P802.11ac/D2.0, January 2012 (22.3.18.5)
Transmit center frequency leakage	IEEE P802.11ac/D2.0, January 2012 (22.3.18.5.2)
Transmitter constellation error	IEEE P802.11ac/D2.0, January 2012 (22.3.18.5.3)
Transmitter modulation accuracy (EVM) test	IEEE P802.11ac/D2.0, January 2012 (22.3.18.5.4)

A.3 MX887032A Specification

Item	Specification
Product Number	MX887032A
Product Name	WLAN 802.11p TX Measurement MX887032A
Use application	The MX887032A WLAN 802.11p TX Measurement option are designed for measuring the transmitter of IEEE802.11p when this application is installed in MU887000A TRX Test module and using either port 3 or 4. Definition of terms. CAL: CAL is the self-calibration function in the measurement instrument. The specifications are applicable after 30 minutes warm up at constant ambient temperature unless otherwise specified.
Product configuration	MX887032A 1 DVD-R 1 Operation Manual 1

Item	Specification
Electrical characteristic	Specifications are for Port 3 and 4 unless otherwise stated.
Common Item	
Measuring Object	WLAN Signal Packet
Frequency Range	715 to 765 MHz 902 to 928 MHz 5725 to 5925 MHz (Required MU887000A-001 option)
RF Power	
Input Setting Range	–65 to +25 dBm (Port 3/4)
Measurement Accuracy	After calibration (at 20 to 30°C) ±0.7 dB (–30 dBm ≤ Level ≤ +25 dBm) ±1.0 dB (–50 dBm ≤ Level < –30 dBm)
Power measurement bandwidths	20/10/ 5MHz
EVM (Modulation Accuracy)	
EVM Measurement Range	–20 to +25 dBm
Residual EVM	
OFDM	Signal Level > –20 dBm, averaged over 20 packets Channel Estimation: FULLPACKET < –40 dB
EVM Data Format	dB or %
Measurement resolution	0.1% or 0.1 dB. All limit checking in dB to 0.1 dB resolution
OFDM EVM Measurement Settings	
Channel estimation	User selection of Long Training Sequence or Full Packet. Default Long Training sequence
User Specified measurement Range	Minimum 16 Max 1000 symbols, default 40
OFDM Pilot tracking	"Phase tracking only" or "Phase and Amplitude Tracking". Default Phase tracking only Peak and Average EVM on all sub-carriers, dB or percentage Peak and Average on each sub-carrier – frequency domain % v sub-carrier EVM v Symbol – time domain % v Symbol number, 1 to max

Item	Specification
OFDM Additional Measurements	
Transmit center frequency tolerance	
Definition	Average frequency of the OFDM carrier signal
Data output format	Hz and ppm
Accuracy	> 1 ms packet ± (Setting frequency × Reference oscillator accuracy + 1 kHz)
Resolution	Hz to 1 Hz, ppm to 1 decimal place
Transmitter center frequency leakage	
Definition	Measurement of the leakage of the center carrier
Data output format	dB
Resolution	dB to two decimal places
Reference specification	
Transmitter Power Levels	IEEE 802.11-2012 (18.3.9.2) (D.2.2)
Transmitter Spectrum Mask	IEEE 802.11-2012 (18.3.9.3) (D.2.3)
Transmit center frequency tolerance	IEEE 802.11-2012 (18.3.9.5)
Symbol Clock frequency tolerance	IEEE 802.11-2012 (18.3.9.6)
Transmitter center frequency leakage	IEEE 802.11-2012 (18.3.9.7.2)
Transmitter spectral flatness	IEEE 802.11-2012 (18.3.9.7.3)
Transmitter modulation accuracy	IEEE 802.11-2012 (18.3.9.7.4) (18.3.9.8)

A.4 MX887033A Specification

Item	Specification
Product Number	MX887033A
Product Name	WLAN 802.11ax TX Measurement MX887033A
Use application	The MX887033A WLAN 802.11ax TX Measurement option are designed for measuring the transmitter of IEEE802.11ax when this application is installed in MU887000A TRX Test module and using either port 3 or 4. IEEE802.11ax (5GHz) require the 6GHz option (MU887000A-001) Definition of terms. CAL: CAL is the self-calibration function in the measurement instrument. The specifications are applicable after 30 minutes warm up at constant ambient temperature unless otherwise specified.
Product configuration	MX887033A 1 DVD-R 1 Operation Manual 1

Item	Specification
Electrical characteristic	Specifications are for Port 3 and 4 unless otherwise stated.
Common Item	
Measuring Object	WLAN Signal Packet
Frequency Range	5GHz Band: (Required MU887000A-001 option) 80MHz BW: 4920 to 5775 MHz 40MHz BW: 4920 to 5795 MHz 20MHz BW: 4920 to 5825 MHz 2.4GHz Band: 40MHz BW: 2412 to 2472 MHz 20MHz BW: 2412 to 2484 MHz
RF Power	
Input Setting Range	–65 to +25 dBm (Port 3/4)
Measurement Accuracy	After calibration (at 20 to 30°C) ±0.7 dB (–30 dBm ≤ Level ≤ +25 dBm) ±1.0 dB (–50 dBm ≤ Level < –30 dBm)
Power measurement bandwidths	80 MHz / 40 MHz / 20 MHz
Capture time	1.34 s
Pre-trigger	1.33 s
Resolution (time domain profile)	5 ns/sample
CCDF	CCDF defined as a percentage of samples against dB, where percentage of samples is normalized to the average power in the ‘gate’, and dB is defined as the relative value of samples greater than the average.
Power distribution value	A single numeric value called the power distribution value defines the number of dB above the average power below which a user defined percentage of the total number of samples falls.

Item	Specification
Spectral profile measurements	
Spectral profile measurement span	±80 MHz
Minimum capture time	50 µs
Input signal measurement range with set RBW (100 kHz)	–27 to +25 dBm
Linearity	CW,RBW = 100 kHz ±0.2 dB (0 to –40dB, ≥ –55dBm)
Resolution	0.1 dB
Measurement bandwidth	100 kHz
EVM (Modulation Accuracy)	
EVM Measurement Range	–20 to +25 dBm
Residual EVM	
≤ 80 MHz	(at 20 to 30°C) Signal Level > –10 dBm, averaged over 20 packets, where each packet is no less than 16 data OFDM symbols long. And for each subcarrier (except Pilots), all data OFDM symbols have same data field pattern. Channel Estimation: FULLPACKET < –40 dB
EVM Data Format	dB or %
Measurement resolution	0.1% or 0.1 dB. All limit checking in dB to 0.1 dB resolution

Item	Specification
OFDM EVM Measurement Settings	
Channel estimation	User selection of Long Training Sequence or Full Packet. Default Long Training sequence
User Specified measurement Range	Minimum 16 Max 1000 symbols, default 40
OFDM Pilot tracking	"Phase tracking only" or "Phase and Amplitude Tracking". Default Phase tracking only Peak and Average EVM on all sub-carriers, dB or percentage Peak and Average on each sub-carrier – frequency domain % v sub-carrier EVM v Symbol – time domain % v Symbol number, 1 to max
OFDM Additional Measurements	
Transmit center frequency tolerance	
Definition	Average frequency of the OFDM carrier signal
Data output format	Hz and ppm
Accuracy	± (Setting frequency × Reference oscillator accuracy + 1 kHz) averaged over 20 packets, where each packet is no less than 16 data OFDM symbols long Channel Estimation: FULLPACKET
Resolution	Hz to 1 Hz, ppm to 1 decimal place
Symbol clock frequency tolerance	
Definition	Frequency error relative to the symbol clock depends on Signal's Guard interval. If GI is 0.8μs, Symbol Clock is (1 / (12.8μs + 0.8μs)) = 73.529 kHz approx. If GI is 1.6μs, Symbol Clock is (1 / (12.8μs + 1.6μs)) = 69.444 kHz approx. If GI is 3.2μs, Symbol Clock is (1 / (12.8μs + 3.2μs)) = 62.500 kHz approx. Measurement averaged over a fully coded OFDM packet with a minimum payload length of 16 symbols.
Data output format	Hz and ppm
Range	±40 ppm
Resolution	ppm to 1 decimal place
User specified measurement range	16-to (define numbers), default 55

Item	Specification
Transmitter center frequency leakage	
Definition	Measurement of the leakage of the center carrier
Data output format	dB
Resolution	dB to two decimal places
Transmitter spectral flatness	
Definition	Measurement of RF sub-carrier power level
Unit of measurement	dB
Reference specification	
Transmit spectrum mask	IEEE P802.11ax/D1.3, June 2017 (28.3.18.1)
Spectral flatness	IEEE P802.11ax/D1.3, June 2017 (28.3.18.2)
Transmit center frequency tolerance	IEEE P802.11ax/D1.3, June 2017 (28.3.18.3)
Symbol clock frequency tolerance	IEEE P802.11ax/D1.3, June 2017 (28.3.18.3)
Modulation accuracy	IEEE P802.11ax/D1.3, June 2017 (28.3.18.4)
Transmit center frequency leakage	IEEE P802.11ax/D1.3, June 2017 (28.3.18.4.2)
Transmitter constellation error	IEEE P802.11ax/D1.3, June 2017 (28.3.18.4.3)
Transmitter modulation accuracy (EVM) test	IEEE P802.11ax/D1.3, June 2017 (28.3.18.4.4)

A.5 MX887040A Specification

Item	Specification
Product Number	MX887040A
Product Name	*Bluetooth* TX Measurement MX887040A
Use application	The MX887040A Bluetooth TX Measurement option is designed for measuring the transmitter performance of Bluetooth (2.0,3.0,4.0) radios when this application is installed in MU887000A TRX Test module and using either port 3 or 4. Definition of terms. CAL: CAL is the self-calibration function in the measurement instrument. Nominal: Represents characteristic performance which is non-warranted. Represents the value of a parameter that is most likely to occur; the expected mean or average. The specifications are applicable after 30 minutes warm up at constant ambient temperature unless otherwise specified.
Product configuration	MX887040A 1 DVD-R 1 Operation Manual 1
Option	MX887040A-001
Product Name	DLE TX Measurement MX887040A-001
Use application	The MX887040A-001 DLE TX Measurement option is designed for measuring the transmitter performance of Bluetooth Low Energy Data Length Extension defined by Bluetooth (4.2) radio. This option requires MX887040A option installed.
Option	MX887040A-002
Product Name	2LE TX Measurement MX887040A-002
Use application	The MX887040A-002 2LE TX Measurement option is designed for measuring the transmitter performance of Bluetooth Low Energy 2 Mbps defined by Bluetooth (5.0) radio. This option requires MX887040A option installed.
Option	MX887040A-003
Product Name	BLR TX Measurement MX887040A-003
Use application	The MX887040A-003 BLR TX Measurement option is designed for measuring the transmitter performance of Bluetooth Long Range defined by Bluetooth (5.0) radio. This option requires MX887040A option installed.

Item	Specification
Electrical characteristic	Specifications are for Port 3 and 4 unless otherwise stated.
Common Item	
Measuring Object	*Bluetooth* Signal Packet (DH-1,3,5 2-DH-1,3,5 3-DH-1,3,5) BLE test packet
Frequency Range	2402 to 2480 MHz
Measurement Mode	'Speed Test' Supports all RF measurements on all packet types 'SIG Standard' Supports RF measurements on selected packet types as per the SIG RF test standard.
RF Power	
Input Setting Range	–65 to +25 dBm (Port 3/4)
Measurement Accuracy	After calibration (at 20 to 30°C) ±0.7 dB (–30 dBm ≤ Level ≤ +25 dBm) (at 20 to 30°C) ±1.0 dB (–50 dBm ≤ Level < –30 dBm)
EDR relative transmit power	
Power measurement rage	–35 to +25 dBm
Measurement	Maximum, minimum, average differential power
Relative Input signal range	Relative Input signal range between the GFSK and π/4DQPSK or 8DSPK sections of the packet.
Power Measurement Bandwidth	1.3 MHz (IF Filter response 'flat'fc±550 kHz)
Maximum Resolution (time domain)	
Resolution	0.01 dB
Bluetooth Modulation	GFSK, π/4DQPSK, 8DPSK

Item	Specification
DEVM (Modulation Accuracy)	
DEVM Measurement Range	–20 to +25 dBm
Residual DEVM	5%
Measurement resolution	0.1%
GFSK Modulation	
Deviation Measurement Range	0 to 350 kHz
Accuracy	Modulation Index = 0.32 Signal level > –20 dBm, averaged over 10 packets Nominal: 1% (+/–0.01 × expected deviation [Hz])
Initial Carrier Frequency tolerance	
Input signal range	–35 to +25 dBm
Initial Frequency Measurement Range	0 to ±150 kHz
Resolution	1 kHz
Carrier-Frequency drift	
Input signal range	–35 to +25 dBm
Frequency drift range	0 to ±200 kHz
Time settings	50 μs, > 2000 μs
EDR Carrier frequency stability	
Measurement range	±100 kHz
Resolution	1 kHz
Accuracy	Signal level > –20 dBm, averaged over 10 packets ± (Setting frequency × Reference oscillator accuracy + 500 Hz)
Displayed results	Initial frequency error ω_i, frequency error ω_o, frequency error $\omega_i + \omega_o$

Item	Specification
EDR Modulation accuracy	
RMS DEVM Range	0 to 30% π/4DQPSK, 0 to 20% 8DPSK
Peak DEVM Range	0 to 50% π/4DQPSK, 0 to 30% 8DPSK
BLE Modulation characteristics	GFSK
Input signal range	–35 to +25 dBm
Frequency deviation measurement range	0 to ±500 kHz peak
Resolution	1 kHz
Accuracy	Modulation Index = 0.5 Signal level > –20 dBm, averaged over 10 packets Nominal: 1% (±0.01 × expected deviation [Hz])
BLE Carrier frequency offset and drift	
Input signal range	–35 to +25 dBm
Frequency measurement range	0 to ±500 kHz
Accuracy	Signal level > –20 dBm, averaged over 10 packets ± (Setting frequency × Reference oscillator accuracy + 500 Hz)
Displayed results	Carrier frequency error, frequency drift, drift rate

Item	Specification
Measurement Capability Summary	Equivalent SIG Test Specification
Output Power	TRM/CA/01/C
Power Control	TRM/CA/03/C
Initial carrier frequency tolerance	TRM/CA/08C
Carrier Frequency drift	TRM/CA/09/C
Modulation Index	TRM/CA/07/C
EDR Carrier frequency stability	TRM/CA/11/C
EDR Modulation accuracy	TRM/CA/11/C
EDR relative transmit power	TRM/CA/10/C
BLE Output power	TRM-LE/CA/BV-01-C
BLE Modulation characteristics	TRM-LE/CA/BV-05-C and TRM-LE/CA/BV-10-C and TRM-LE/CA/BV-13-C
BLE Stable Modulation characteristics	TRM-LE/CA/BV-09-C and TRM-LE/CA/BV-11-C
BLE Carrier frequency offset and drift	TRM-LE/CA/BV-06-C and TRM-LE/CA/BV-12-C and TRM-LE/CA/BV-14-C

A.6 MX887050A Specification

Item	Specification
Product Number	MX887050A
Product Name	Short Range Wireless Average Power and Frequency Measurement MX887050A
Use application	The MX887050A are designed for measuring the transmitter power and frequency performance of Bluetooth (2.0,3.0,4.0) and IEEE802.11 b, g, a, n(1X1) HT 20/40MHz radios when this application is installed in MU887000A TRX Test module and using either port 3 or 4. Definition of terms. CAL: CAL is the self-calibration function in the measurement instrument. The specifications are applicable after 30 minutes warm up at constant ambient temperature unless otherwise specified.
Product configuration	MX887050A 1 DVD-R 1 Operation Manual 1

Item	Specification
Electrical characteristic	Specifications are for Port 3 and 4 unless otherwise stated.
RF Power (CW and continuously Modulated)	
Input Setting Range	–65 to +25 dBm (Port 3/4)
Frequency Range	2.4GHz Band: 2402 to 2484 MHz 5GHz Band: 4920 to 5825 MHz (Require MU887000A-001 option)
Measurement Accuracy	After calibration (400MHz ≤ Frequency ≤ 3800MHz) 10°C ≤ Temperature ≤ 40°C ±0.7 dB (–30 dBm ≤ Level ≤ +25 dBm) ±0.9 dB (–55 dBm ≤ Level < –30 dBm) ±1.1 dB (–65 dBm≤ Level < –55 dBm) (3800 MHz ≤ Frequency ≤ 6000 MHz) 20°C ≤ Temperature ≤ 30°C ±0.7 dB (–30 dBm ≤ Level ≤ +25 dBm) ±0.9 dB (–55 dBm ≤ Level < –30 dBm) ±1.1 dB (–65 dBm ≤ Level < –55 dBm)
Linearity	CW, RBW = 100 kHz ±0.2 dB (0 to –40 dB, ≥ –55 dBm)
Frequency (CW)	
Input signal range	–35 to +25 dBm
Frequency measurement range	0 to ±500 kHz
Accuracy	± (Setting frequency × Reference oscillator accuracy + 500 Hz)

Appendix B Glossary

This chapter explains some of the acronyms and abbreviations that are used throughout this manual.

Term	Meaning
ACK	Acknowledgment
ADC	Analog-to-digital convertor
ARB	Arbitrary Waveform Generator
BLE	*Bluetooth* Low Energy
BR	Basic Rate (*Bluetooth*)
BT	*Bluetooth*
DEVM	Differential Error Vector Magnitude (*Bluetooth*)
DUT	Device Under Test
EDR	Enhanced Data Rate (*Bluetooth*)
EVM	Error Vector Magnitude
FPGA	Field-Programmable Grip Array
GNSS	Global Navigation Satellite System
IQ	In-band and Quadrature-band
LO	Local Oscillator
MCS	Modulation and Coding Scheme
ME	Measurement Engine
OBW	Occupied band width
PA	Power Amplifier
PLCP	Physical Layer Convergence Protocol
PPDU	PLCP Protocol Data Unit
SCPI	Standard Commands for Programmable Instruments
SIG	Special Interest Group (*Bluetooth*)
VSA	Vector Signal Analyzer
VSG	Vector Signal Generator
WLAN	Wireless Local Area Network