

# R&S<sup>®</sup>SMM-K60/-K117

## Bluetooth<sup>®</sup>Enhanced Data Rate, Bluetooth<sup>®</sup>5.x

### User Manual



1179205402  
Version 06

**ROHDE & SCHWARZ**  
Make ideas real



This document describes the following software options:

- R&S®SMM-K60 Bluetooth EDR (1441.1918.xx)
- R&S®SMM-K117 Bluetooth 5.x (1441.1799.xx)

This manual describes firmware version FW 5.30.047.xx and later of the R&S®SMM100A.

© 2023 Rohde & Schwarz

Muehldorfstr. 15, 81671 Muenchen, Germany

Phone: +49 89 41 29 - 0

Email: [info@rohde-schwarz.com](mailto:info@rohde-schwarz.com)

Internet: [www.rohde-schwarz.com](http://www.rohde-schwarz.com)

Subject to change – data without tolerance limits is not binding.

R&S® is a registered trademark of Rohde & Schwarz GmbH & Co. KG.

The Bluetooth® word mark and logos are registered trademarks owned by Bluetooth SIG, Inc. and any use of such marks by Rohde & Schwarz is under license.

All other trademarks are the properties of their respective owners.

1179.2054.02 | Version 06 | R&S®SMM-K60/-K117

The following abbreviations are used throughout this manual: R&S®SMM100A is abbreviated as R&S SMM, R&S®WinIQSIM2™ is abbreviated as R&S WinIQSIM2; the license types 02/03/07/11/13/16/12 are abbreviated as xx.

# Contents

<b>1</b>	<b>Welcome to the Bluetooth options.....</b>	<b>7</b>
1.1	Key features.....	7
1.2	Accessing the Bluetooth dialog.....	8
1.3	What's new.....	8
1.4	Documentation overview.....	9
1.4.1	Getting started manual.....	9
1.4.2	User manuals and help.....	9
1.4.3	Service manual.....	9
1.4.4	Instrument security procedures.....	9
1.4.5	Printed safety instructions.....	10
1.4.6	Data sheets and brochures.....	10
1.4.7	Release notes and open source acknowledgment (OSA).....	10
1.4.8	Application notes, application cards, white papers, etc.....	10
1.4.9	Videos.....	10
1.5	Scope.....	11
1.6	Notes on screenshots.....	11
<b>2</b>	<b>About the Bluetooth options.....</b>	<b>12</b>
2.1	Required options.....	12
2.2	About Bluetooth BR/EDR.....	12
2.2.1	Bluetooth packet types for BR/EDR.....	13
2.2.2	Bluetooth transport modes.....	16
2.2.3	Packet structure and fields.....	16
2.2.4	Bluetooth modulation schemes.....	19
2.3	About Bluetooth LE.....	19
2.3.1	Packet formats for LE.....	20
2.3.2	Packet types for LE.....	22
2.3.3	Packet structure and fields.....	23
2.3.4	Modulation scheme.....	26
2.3.5	Direction finding.....	27
<b>3</b>	<b>Bluetooth configuration and settings.....</b>	<b>32</b>
3.1	General settings.....	32

<b>3.2</b>	<b>Dirty transmitter test settings.....</b>	<b>35</b>
<b>3.3</b>	<b>Channel settings - BR/EDR.....</b>	<b>40</b>
<b>3.4</b>	<b>Packet configuration - BR/EDR.....</b>	<b>41</b>
<b>3.5</b>	<b>Channel settings - LE.....</b>	<b>49</b>
<b>3.6</b>	<b>Event / Frame configuration - LE.....</b>	<b>54</b>
3.6.1	Advertising event / frame settings.....	56
3.6.2	Data event settings.....	58
3.6.3	Channel table settings.....	61
<b>3.7</b>	<b>Packet configuration - LE.....</b>	<b>62</b>
3.7.1	General packet configuration.....	63
3.7.2	Header configuration.....	63
3.7.3	Main payload configuration dialog.....	67
3.7.4	Additional payload configuration dialogs.....	83
<b>3.8</b>	<b>Test packet configuration - LE.....</b>	<b>88</b>
<b>4</b>	<b>Signal control and signal characteristics.....</b>	<b>90</b>
<b>4.1</b>	<b>Filter/clipping settings.....</b>	<b>90</b>
4.1.1	Filter settings.....	91
4.1.2	Modulation settings.....	93
4.1.3	Clipping settings.....	94
<b>4.2</b>	<b>Power ramping settings.....</b>	<b>96</b>
<b>4.3</b>	<b>Trigger settings.....</b>	<b>97</b>
<b>4.4</b>	<b>Marker settings.....</b>	<b>102</b>
<b>4.5</b>	<b>Clock settings.....</b>	<b>104</b>
<b>4.6</b>	<b>Local and global connectors settings.....</b>	<b>105</b>
<b>5</b>	<b>Remote-control commands.....</b>	<b>106</b>
<b>5.1</b>	<b>General commands.....</b>	<b>107</b>
<b>5.2</b>	<b>Dirty transmitter configuration.....</b>	<b>112</b>
<b>5.3</b>	<b>Channel configuration commands - BR/EDR.....</b>	<b>119</b>
<b>5.4</b>	<b>Packet configuration commands - BR/EDR.....</b>	<b>120</b>
<b>5.5</b>	<b>Channel configuration commands - LE.....</b>	<b>129</b>
<b>5.6</b>	<b>Event and frame configuration commands - LE.....</b>	<b>134</b>
<b>5.7</b>	<b>Packet configuration commands - LE.....</b>	<b>144</b>
5.7.1	General configuration.....	150

5.7.2	Header configuration.....	151
5.7.3	Payload configuration.....	154
<b>5.8</b>	<b>Test packet configuration commands - LE.....</b>	<b>183</b>
<b>5.9</b>	<b>Filter/clipping commands.....</b>	<b>185</b>
<b>5.10</b>	<b>Power ramping commands.....</b>	<b>192</b>
<b>5.11</b>	<b>Trigger commands.....</b>	<b>194</b>
<b>5.12</b>	<b>Marker commands.....</b>	<b>199</b>
<b>5.13</b>	<b>Clock commands.....</b>	<b>203</b>
	<b>Glossary: Specifications.....</b>	<b>205</b>
	<b>List of commands.....</b>	<b>206</b>
	<b>Index.....</b>	<b>212</b>



# 1 Welcome to the Bluetooth options

The R&S SMM-K60 is a firmware application that adds functionality to generate signals in accordance with the Bluetooth version 4.2.

Option R&S SMM-K117 adds support for Bluetooth LE signals according to the core specification v5.4 for Bluetooth wireless technology. This option is an extension of R&S SMM-K60.

This user manual contains a description of the functionality that the application provides, including remote control operation.

All functions not discussed in this manual are the same as in the base unit and are described in the R&S SMM100A user manual. The latest version is available at:

[www.rohde-schwarz.com/manual/SMM100A](http://www.rohde-schwarz.com/manual/SMM100A)

## Installation

You can find detailed installation instructions in the delivery of the option or in the R&S SMM100A service manual.

## 1.1 Key features

Option R&S SMM-K60 provides Bluetooth signals for basic rate (BR) and enhanced data rate (EDR) burst types. In addition, it provides also low energy (LE) signals limited to LE 1 Msymbol/s physical layer.

The following BR and EDR features are supported within **R&S SMM-K60**:

- Support for three transport modes, the ACL+EDR, SCO, eSCO+EDR transport modes.
- Support of all packet types for both the basic rate and the enhanced data rate modes.
- Generation of signals with up to 5238 frames sequence length.
- Configuration of the packet contents with a convenient packet editor or all data packets, both with optional data whitening.
- Generation of signals in accordance to the "Dirty Transmitter Test" specification for both, the basic and enhanced data rates. The test enables you to change the start phase, the frequency drift rate and the frequency drift deviation.
- Power ramp control with possibilities to choose ramp time, rise and fall offset
- Configuration of the clipping, filter and modulation settings

The following LE features are supported within R&S SMM-K60:

- Support for two channel types, the "Advertising" and "Data" channel types.
- Support of all Bluetooth packet types for LE 1 Msymbol/s physical layer (LE 1M PHY).
- Convenient packet editor for all supported packet types including optional data whitening.

- Dirty transmitter test, compliant to the RF test specification with options to change start phase, frequency drift rate and frequency drift deviation.
- Support of CRC corruption for every 2nd packet
- Power ramp control with configurable ramp time, rise and fall offsets.
- Clipping, filter and modulation settings supported.

The following LE features are supported within **R&S SMM-K117**:

- Support for two channel types, the "Advertising" and "Data" channel types.
- Support of all Bluetooth packet types for uncoded LE 2 Msymbol/s physical layer (LE 2M PHY)
- Support of all Bluetooth packet types for LE coded 1 Msymbol/s physical layer (LE coded PHY)
- Support of CRC corruption for every 2nd packet
- Convenient packet editor for all supported packet types including optional data whitening.
- Dirty transmitter test, compliant to the RF test specification with options to change start phase, frequency drift rate, frequency drift deviation, and modulation index mode.
- Support of Bluetooth Direction Finding using Constant Tone Extension methods Angle of Arrival or Angle of Departure

## 1.2 Accessing the Bluetooth dialog

### To open the dialog with Bluetooth settings

- ▶ In the block diagram of the R&S SMM100A, select "Baseband > Bluetooth".

A dialog box opens that displays the provided general settings.

The signal generation is not started immediately. To start signal generation with the default settings, select "State > On".

## 1.3 What's new

This manual describes firmware version FW 5.30.047.xx and later of the R&S®SMM100A.

Compared to the previous version, it provides the new features listed below:

- Time-based triggering, see "[Time Based Trigger](#)" on page 99 and "[Trigger Time](#)" on page 99.
- Editorial changes



## 1.4 Documentation overview

This section provides an overview of the R&S SMM100A user documentation. Unless specified otherwise, you find the documents at:

[www.rohde-schwarz.com/manual/smm100a](http://www.rohde-schwarz.com/manual/smm100a)

### 1.4.1 Getting started manual

Introduces the R&S SMM100A and describes how to set up and start working with the product. Includes basic operations, typical measurement examples, and general information, e.g. safety instructions, etc. A printed version is delivered with the instrument.

### 1.4.2 User manuals and help

Separate manuals for the base unit and the software options are provided for download:

- **Base unit manual**  
Contains the description of all instrument modes and functions. It also provides an introduction to remote control, a complete description of the remote control commands with programming examples, and information on maintenance, instrument interfaces and error messages. Includes the contents of the getting started manual.
- **Software option manual**  
Contains the description of the specific functions of an option. Basic information on operating the R&S SMM100A is not included.

The contents of the user manuals are available as help in the R&S SMM100A. The help offers quick, context-sensitive access to the complete information for the base unit and the software options.

All user manuals are also available for download or for immediate display on the Internet.

### 1.4.3 Service manual

Describes the performance test for checking compliance with rated specifications, firmware update, troubleshooting, adjustments, installing options and maintenance.

The service manual is available for registered users on the global Rohde & Schwarz information system (GLORIS):

<https://gloris.rohde-schwarz.com>

### 1.4.4 Instrument security procedures

Deals with security issues when working with the R&S SMM100A in secure areas. It is available for download on the internet.

### 1.4.5 Printed safety instructions

Provides safety information in many languages. The printed document is delivered with the product.

### 1.4.6 Data sheets and brochures

The data sheet contains the technical specifications of the R&S SMM100A. It also lists the options and their order numbers and optional accessories.

The brochure provides an overview of the instrument and deals with the specific characteristics.

See [www.rohde-schwarz.com/brochure-datasheet/smm100a](http://www.rohde-schwarz.com/brochure-datasheet/smm100a)

### 1.4.7 Release notes and open source acknowledgment (OSA)

The release notes list new features, improvements and known issues of the current firmware version, and describe the firmware installation.

The software makes use of several valuable open source software packages. An open-source acknowledgment document provides verbatim license texts of the used open source software.

See [www.rohde-schwarz.com/firmware/smm100a](http://www.rohde-schwarz.com/firmware/smm100a)

### 1.4.8 Application notes, application cards, white papers, etc.

These documents deal with special applications or background information on particular topics.

See [www.rohde-schwarz.com/application/smm100a](http://www.rohde-schwarz.com/application/smm100a)

### 1.4.9 Videos

Find various videos on Rohde & Schwarz products and test and measurement topics on YouTube: <https://www.youtube.com/@RohdeundSchwarz>



On the menu bar, search for your product to find related videos.

HOME VIDEOS SHORTS PLAYLISTS COMMUNITY CHANNELS ABOUT

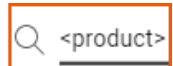


Figure 1-1: Product search on YouTube

## 1.5 Scope



Tasks (in manual or remote operation) that are also performed in the base unit in the same way are not described here.

In particular, it includes:

- Managing settings and data lists, like saving and loading settings, creating and accessing data lists, or accessing files in a particular directory.
- Information on regular trigger, marker and clock signals and filter settings, if appropriate.
- General instrument configuration, such as checking the system configuration, configuring networks and remote operation
- Using the common status registers

For a description of such tasks, see the R&S SMM100A user manual.

## 1.6 Notes on screenshots

When describing the functions of the product, we use sample screenshots. These screenshots are meant to illustrate as many as possible of the provided functions and possible interdependencies between parameters. The shown values may not represent realistic usage scenarios.

The screenshots usually show a fully equipped product, that is: with all options installed. Thus, some functions shown in the screenshots may not be available in your particular product configuration.

## 2 About the Bluetooth options

The R&S SMM100A provides you with the ability to generate signals in accordance with the core specification 5.4 for Bluetooth wireless technology.

This section lists required options and provides background information on basic terms and principles used in Bluetooth technology.

### 2.1 Required options

The basic equipment layout for generating Bluetooth signals includes the:

- Baseband Generator(R&S SMM-B9)
- Frequency option (e.g. R&S SMM-B1006)
- Baseband realtime extension (R&S SMM-K520)
- Option Bluetooth EDR (R&S SMM-K60)
- Option Bluetooth 5.x (R&S SMM-K117)

You can generate signals via play-back of waveform files at the signal generator. To create the waveform file using R&S WinIQSIM2, you do not need a specific option.

To play back the waveform file at the signal generator, you have two options:

- Install the R&S WinIQSIM2 option of the digital standard, e.g. R&S SMM-K255 for playing LTE waveforms
- If supported, install the real-time option of the digital standard, e.g. R&S SMM-K55 for playing LTE waveforms

For more information, see data sheet.

### 2.2 About Bluetooth BR/EDR

The frequency band defined for Bluetooth devices is the unlicensed 2.4 GHz Industrial, Scientific and Medical (ISM) frequency band.

**Table 2-1: Operating band**

Regulatory range	RF channels k and center frequencies f
2400.0 MHz to 2483.5 MHz	$k = 0$ to $78$ , $f = k * 1 \text{ MHz} + 2402 \text{ MHz}$

Two modulation modes are used for Bluetooth: the mandatory basic rate (BR) and the optional enhanced data rate (EDR). The BR mode uses binary FM modulation and provides a data rate of 1 Mbps. The EDR mode uses two types of PSK modulation, the  $\pi/4$ -DQPSK or 8DPSK, and achieves data rates of 2 Mbps and 3 Mbps, respectively. All modulations schemes have the symbol rate equal to 1 Msymbol/s.

A time division duplex (TDD) scheme for duplex transmission is defined for both modes.

The following sections describe signal characteristics in detail:

- [Bluetooth packet types for BR/EDR](#)..... 13
- [Bluetooth transport modes](#)..... 16
- [Packet structure and fields](#)..... 16
- [Bluetooth modulation schemes](#)..... 19

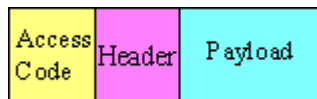
## 2.2.1 Bluetooth packet types for BR/EDR

### 2.2.1.1 ACL packets

The ACL packets are used for asymmetric links and they contain user data or control data. The table and the figures below give an overview of the ACL packets and their structure.

**Table 2-2: ACL packet - basic rate**

Type	Payload Header (bytes)	User Payload (bytes)	FEC	CRC	Slot number
DM1	1	0-17	2/3	Yes, 16-bit	1
DH1		0-27	no		
DM3	2	0-121	2/3		5
DH3		0-183	no		
DM5		0-224	2/3		
DH5		0-339	no		
AUX1	1	0-29		no	



**Figure 2-1: Packet structure of ACL packets - basic rate**

**Table 2-3: ACL packets - enhanced rate**

Type	Payload Header (bytes)	User Payload (bytes)	FEC	CRC	Slot number
2-DH1	2	0-54	no	Yes, 16-bit	1
2-DH3		0-367			3
2-DH5		0-679			5
3-DH1		0-83			1
3-DH3		0-552			3
2-DH5		0-1021			5



Figure 2-2: Packet structure of ACL packets - enhanced data rate

### 2.2.1.2 SCO and eSCO packets

The SCO and eSCO packets are used for symmetric links. The SCO packets are used for 64 kb/s speech transmission and for transparent synchronous data. The eSCO packets are also used for 64kb/s speech transmission and transparent data at 64 kb/s but also at other rates.

The tables and the figures below give an overview of the SCO and eSCO packets and their structure.

Table 2-4: SCO packets

Type	Payload Header (bytes)	User Payload (bytes)	FEC	CRC	Slot number
HV1	n.a.	10	1/3	no	n.a.
HV2		20	2/3		
HV3		30			
DV	1 (data only)	10+(0-9)	2/3 (data only)	Yes, 16-bit (data only)	

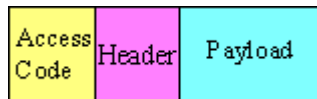


Figure 2-3: Packet structure SCO packets

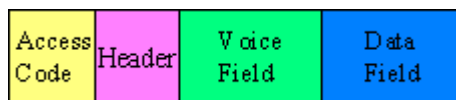


Figure 2-4: Packet structure SCO packets (data only)

Table 2-5: eSCO packets - basic rate

Type	Payload Header (bytes)	User Payload (bytes)	FEC	CRC	Slot number
EV3	n.a.	1-30	no	Yes, 16-bit (Data only)	1
EV4		1-120	2/3		3
EV5		1-180	no		3

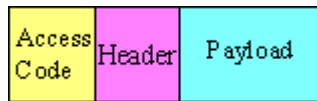


Figure 2-5: Packet structure eSCO packets - basic rate

Table 2-6: eSCO packets - basic rate

Type	Payload Header (bytes)	User Payload (bytes)	FEC	CRC	Slot number
2-EV3	n.a.	1-60	no	Yes, 16-bit	1
2-EV5		1-360			3
3-EV3		1-90			1
3-EV5		1-540			3



Figure 2-6: Packet structure eSCO packets - enhanced data rate

### 2.2.1.3 Link control packets for ACL, SCO, eSCO transport modes

There are some common kinds of packet types. An overview of these packet types is given in the table below.

Table 2-7: Common link control packets

Transport modes	Type	Payload Header (bytes)	FEC	CRC	Application
SCO, eSCO, ACL	ID	n.a.	n.a.	n.a.	Paging, inquiry, response
SCO, eSCO, ACL	NULL				Carries Link information to the source, e.g. about successfully received signal (ARQN) or the state of the receiving buffer (FLOW)
SCO, eSCO, ACL	POLL				Similar to NULL packet, used by the Central to poll the Peripheral devices, must be confirmed
SCO, ACL	FHS	18	2/3	Yes	Page Central response, inquiry response, in roll switch

Table 2-8: Common link control packets: packet structure

Packet Type ID	Packet Types NULL and PULL	Packet Types FHS
Access Code (DAK or IAC)	Access Code Header	Access Code Header Payload

## 2.2.2 Bluetooth transport modes

There are three different transport modes defined in the Bluetooth core specification, each of them with special applications:

- Synchronous connection-oriented (SCO)  
The SCO transport mode is used for a symmetric point-to-point link establishment between a Central and a specific Peripheral in the piconet.
- Extended synchronous connection-oriented (eSCO)  
The eSCO transport mode is used for a symmetric or asymmetric, point-to-point link establishment between the Central and a specific Peripheral.
- Asynchronous connection less (ACL)  
The ACL transport mode is used for a point-to-multipoint link establishment between the Central and all Peripheral participating on the piconet.

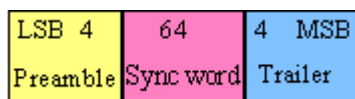
There are some common transmitted packets used by all transport modes and some specific packets defined for each transport mode.

## 2.2.3 Packet structure and fields

Almost all Bluetooth transmitted packets have standard format and consist of the access code, the header and the payload with useful information. The exceptions are the ID packet which consists of the access code only and NULL and POLL packets which carry only the access code and the header.

### 2.2.3.1 Access code

The access code is used for synchronization, DC offset compensation and identification. The fields of the access code are shown in the figure below and their meaning is explained in the table below.



**Table 2-9: The access code fields**

Field	Description	Packets
Preamble	A fixed zero-one pattern of 4 symbols, used to facilitate DC compensation	All packets
Sync word	A 64-bit code word derived from a 24-bit address, improves timing acquisition	All packets
Trailer	A fixed zero-one pattern of four symbols, extended DC compensation	All packets, except ID

### 2.2.3.2 Header

The header contains link control information. The fields of the header are shown in the figure and their meaning is explained in the table below.



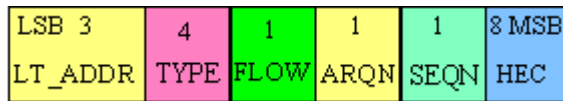


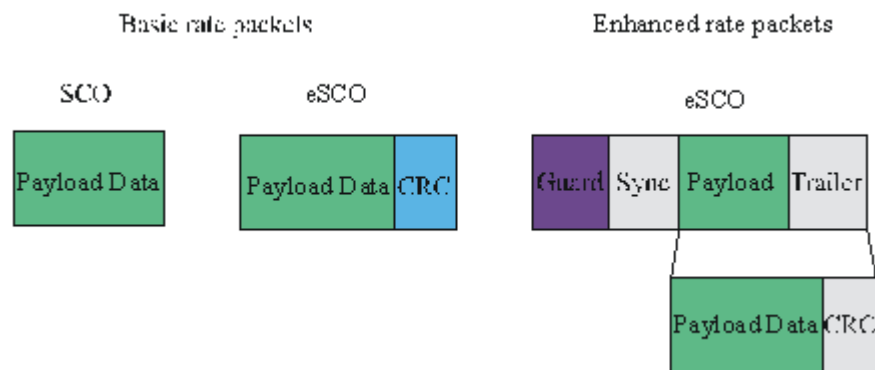
Table 2-10: The header fields

Field	Description	Packets
LT_ADDR	Logical transport address, indicates the destination Peripheral for a packet in a Central-to-Peripheral transmission slot and the source Peripheral for a Peripheral-to-Central transmission slot	
TYPE	Type code, specifies which packet type is used	
FLOW	Flow control, used for flow control of packets over the ACL logical transport. When the RX buffer in the recipient is full, a STOP indication must be returned. When the RX buffer can accept data, a "Go" indication must be returned.	All packets, except ID
ARQN	Automatic repeat request number, acknowledgement indication, used to inform the source of a successful transfer of payload data with CRC can be positive acknowledged ACK or negative acknowledged NAK,	
SEQN	Sequential numbering scheme to order the data packet stream	
HEC	Header-error-check to check the header integrity	

### 2.2.3.3 Payload format

The payload structure depends on the type of the data field and the data rate. Two fields are defined in the payload: the synchronous data field and the asynchronous data field. The ACL packets only have the asynchronous data field and the SCO and eSCO packets only have the synchronous data field. The exception is DV of SCO transport mode which has both data fields, synchronous and asynchronous.

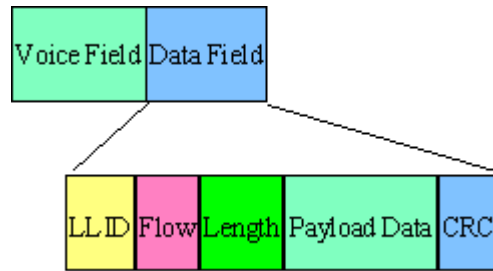
#### Synchronous data fields



#### Asynchronous data fields



### Synchronous and asynchronous data fields

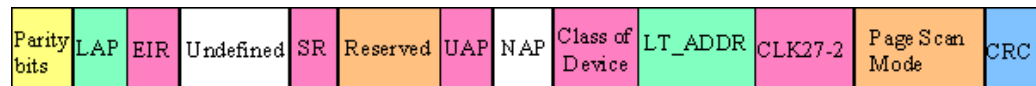


The meaning of some payload fields is given in the table below.

**Table 2-11: The payload fields**

Field	Description
CRC	The cyclic redundancy error check
Guard, sync	The guard time and synchronization sequence, used for physical layer change of modulation scheme
LLID	The logical link identifier, specifies the logical link
Flow	Field which controls the flow on the logical channels

The payload format and content of the FHS packet are different from other packets. The fields of the FHS packet are shown in the figure below and their meaning is explained in the table below.



**Table 2-12: The payload fields for the FHS packet**

Field	Description
Parity bits	Form the first part of the sync word of the access code of the device that sends the FHS packet
LAP	Contains the lower address part of the device that sends the FHS packet
EIR	An extended inquiry response, provides miscellaneous information during the inquiry response procedure
Undefined	Reserved for future use and must be set to zero
SR	The scan repetition field, indicates the interval between two consecutive page scan windows
Reserved	Must be set to 10
UAP	Contains the upper address part of the device that sends the FHS packet
NAP	Contains the non-significant address part of the device that sends the FHS packet
Class of device	Contains the class of device of the device that sends the FHS packet. This field is defined in Bluetooth assigned numbers.
LT_ADDR	Contains the logical transport address

Field	Description
CLK27-2	Contains the value of the native clock of the device that sends the FHS packet, sampled at the beginning of the transmission of the access code of this FHS packet
Page scan mode	Indicates which scan mode is used by default by the sender of the FHS packet

## 2.2.4 Bluetooth modulation schemes

The modulation used for the basic data rate packets is GFSK (Gaussian Frequency Shift Keying) with a bandwidth bit period product  $BT = 0.5$ . The modulation index is between 0.28 and 0.35.

The modulation scheme used for enhanced data rate packets changes within the packet. The access code and packet header have a GFSK modulation scheme and are transmitted with the basic rate 1 Mbps. The subsequent synchronization sequence, payload and trailer sequence have a PSK type of modulation and are transmitted with a data rate of 2 Mbps or optionally 3 Mbps.

The PSK modulation, namely  $\pi/4$  rotated differential encoded quaternary phase shift keying ( $\pi/4$ -DQPSK) is defined for the 2 Mbps transmission.

The PSK modulation, namely differential encoded 8-ary phase shift keying (8DPSK), is defined for the 3 Mbps transmission.

The modulation types and corresponding packet types are given in the table below.

**Table 2-13: The modulation types and corresponding packet types**

Modulation type	Packet types
GFSK	ID, NULL, POLL, FHS, DM1, DH1, DM3, DH3, DM5, DH5, AUX1, HV1, HV2, HV3, DV, EV3, EV4, EV5
GFSK + $\pi/4$ -DQPSK	2-DH1, 2-DH3, 2-DH5, 2-EV3, 2-EV5
GFSK + 8DPSK	3-DH1, 3-DH3, 3-DH5, 3-EV3, 3-EV5

## 2.3 About Bluetooth LE

The R&S SMM100A provides you with the ability to generate signals in accordance with the Low Energy (LE) specification for Bluetooth wireless technology.

Bluetooth LE provides data transfer from low-power devices running on the smallest of batteries to a larger device, such as a PC, a mobile phone, or a PDA. Bluetooth LE establishes a connection, e.g. to a wristwatch, a heart rate sensor, or a data transfer from a digital camera. The generated packets do not support audio content.

A time division duplex (TDD) scheme for duplex transmission is defined. The frequency band defined for Bluetooth devices is the unlicensed 2.4 GHz "Industrial, Scientific and Medical" (ISM) frequency band.

Table 2-14: Operating band

Regulatory range	RF channels k and center frequencies f
2400.0 MHz to 2483.5 MHz	k = 0 to 39, f = k * 2 MHz + 2402 MHz

Table 2-15: Channel index

RF channel	RF center frequency in MHz	Data channel index	Advertising channel index
0	2402	-	37
1 to 11	2404 to 2424	0 to 10	-
12	2426	-	38
13 to 38	2428 to 2478	11 to 36	-
39	2480	-	39



Figure 2-7: RF channels

red = advertising channels (primary)  
 blue = data channels and secondary advertising channels

The core specification of Bluetooth wireless technology defines the limits of output power level at the maximum power setting. The minimum output power is limited to -20 dBm. The maximum output power for LE is limited to 10 dBm.

The maximum output power for LE is limited to 20 dBm.

The following sections describe signal characteristics in detail:

- Packet formats for LE.....20
- Packet types for LE.....22
- Packet structure and fields.....23
- Modulation scheme.....26
- Direction finding.....27

### 2.3.1 Packet formats for LE

#### Packet formats for LE uncoded PHY

The following packet format is defined for the LE uncoded PHYs and is used for both advertising channel packets and data channel packets.

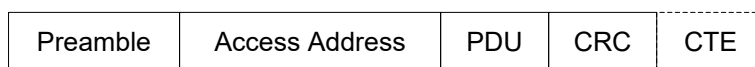


Figure 2-8: LE uncoded PHY packet format

Each packet consists of four mandatory fields: preamble, access address, PDU, and CRC. For Bluetooth [Direction finding](#), the optional field Constant Tone Extension (CTE) is added at the end.

**Table 2-16: Packet format for LE uncoded PHY**

Physical layer	Preamble	Access address	PDU	CRC	CTE
LE 1 Msymbol/s	1 octet	4 octets	2 to 257 octets	3 octets	16 $\mu$ s to 160 $\mu$ s
LE 2 Msymbol/s	2 octets	4 octets	2 to 257 octets	3 octets	16 $\mu$ s to 160 $\mu$ s

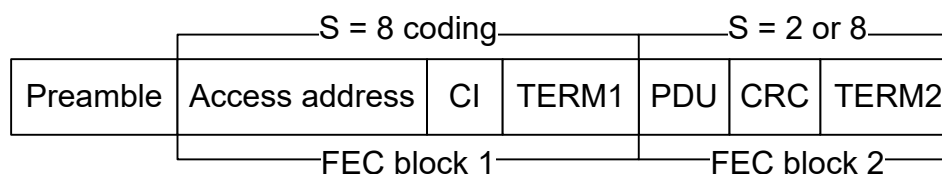
The preamble is transmitted first, followed by the access address, followed by the PDU followed by the CRC and optionally followed by CTE. The entire packet is transmitted at the same symbol rate. Option R&S SMM-K60 supports LE uncoded 1 Msymbol/s (LE 1M) physical layer (PHY).

Option R&S SMM-K117 supports optional modulation scheme LE uncoded 2 Msymbol/s (LE 2M) PHY.

Packets take between 44  $\mu$ s and 2120  $\mu$ s to transmit. The period extends by an additional 16  $\mu$ s to 160  $\mu$ s, if CTE is active.

#### Packet formats for LE coded PHY

The following packet format is defined for the LE coded PHY and is used for both advertising channel packets and data channel packets.



**Figure 2-9: LE coded PHY packet format**

Each packet consists of the preamble, FEC block 1, and FEC block 2. The preamble is not coded. The FEC block 1 consists of three fields: access address, coding indicator (CI), and TERM1. These fields use the S=8 coding scheme. The CI field determines which coding scheme is used for FEC block 2. The FEC block 2 consists of three fields: PDU, CRC, and TERM2. These fields use either the S=2 or S=8 coding scheme, depending on the value of the CI field.

The entire packet is transmitted with 1 Msymbol/s modulation. The following table captures the size and duration of the data packet fields.

**Table 2-17: Packet format for LE coded PHY**

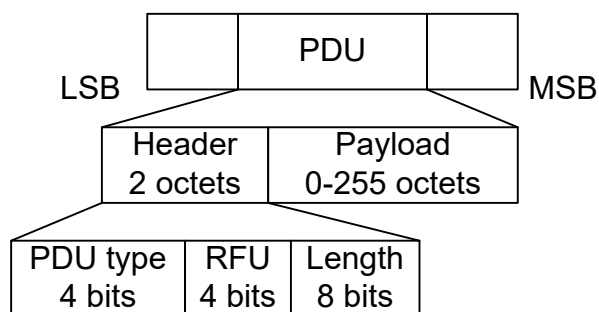
	Pream-ble	Access address	CI	TERM1	PDU	CRC	TERM2
Number of uncoded bits	80	32	2	3	16 - 2056	24	3
Duration in $\mu$ s for S=8 coding	80	256	16	24	128 - 16448	192	24
Duration in $\mu$ s for S=2 coding	80	256	16	24	32 - 4112	48	6

Packets take between 462  $\mu$ s and 17040  $\mu$ s to transmit.

### 2.3.2 Packet types for LE

#### Test packet types

The test packet PDU is subdivided into a PDU header and the payload field. The PDU header indicates the payload content type and the payload length expresses in octets. RFU field means reserved for future use.



LE test packets are described in the "Air Interface Packets" section of core specification for Bluetooth wireless technology, volume 6, part B.

#### Advertising channel packet types

The advertising channel PDU has a 16-bit header and a variable size payload. The header fields of the advertising channel PDU are as shown in "Header" on page 23.

**Table 2-18: Advertising packet types:**

ADV_IND	SCAN_REQ
ADV_DIRECT_IND	SCAN_RSP
ADV_NONCONN_IND	CONNECT_IND
ADV_SCAN_IND	

**Table 2-19: Additional advertising packet types within R&S SMM-K117:**

ADV_EXT_IND	AUX_SCAN_REQ
AUX_ADV_IND	AUX_SCAN_RSP
AUX_CHAIN_IND	AUX_CONNECT_REQ
AUX_SYNC_IND	AUX_CONNECT_RSP

#### Data channel packet types

The data channel PDU has a 16-bit header, a variable size payload, and can include a message integrity check (MIC) field as shown in "Header" on page 26.

The MIC field is not included in an unencrypted link layer (LL) connection, or in an encrypted LL connection with a data channel PDU with a zero length payload. The MIC

field is included in an encrypted LL connection, with a data channel PDU with a non-zero length payload. The MIC calculation is specified in the section 1 of core specification for Bluetooth wireless technology, volume 6, part E.

Besides the data packet type, instrument supports the following CONTROL\_DATA packet types.

**Table 2-20: Control data packet types**

Opcode	CONTROL_DATA	Opcode	CONTROL_DATA
0x00	LL_CONNECTION_UPDATE_IND	0x07	LL_UNKNOWN_RSP
0x01	LL_CHANNEL_MAP_IND	0x08	LL_FEATURE_REQ
0x02	LL_TERMINATE_IND	0x09	LL_FEATURE_RSP
0x03	LL_ENC_REQ	0x0A	LL_PAUSE_ENC_REQ
0x04	LL_ENC_RSP	0x0B	LL_PAUSE_ENC_RSP
0x05	LL_START_ENC_REQ	0x0C	LL_VERSION_IND
0x06	LL_START_ENC_RSP	0x0D	LL_REJECT_IND

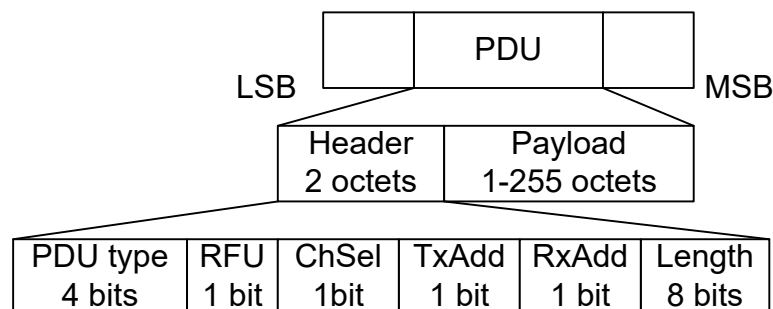
**Table 2-21: Additional control data packet types within R&S SMM-K117:**

Opcode	CONTROL_DATA	Opcode	CONTROL_DATA
0x0E	LL_PERIPHERAL_FEAT_REQ	0x14	LL_LENGTH_REQ
0x0F	LL_CONNECTION_PARAM_REQ	0x15	LL_LENGTH_RSP
0x10	LL_CONNECTION_PARAM_RSP	0x16	LL_PHY_REQ
0x11	LL_REJECT_EXT_IND	0x17	LL_PHY_RSP
0x12	LL_PING_REQ	0x18	LL_PHY_UPDATE_IND
0x13	LL_PING_RSP	0x19	LL_MIN_USED_CHANNELS_IND

### 2.3.3 Packet structure and fields

#### 2.3.3.1 Advertising channel packet structure

##### Header



- The possible **PDU types**, indicated in the header of advertising channel PDU, are listed in the previous tables, see [Table 2-18](#).

The following table shows which channels are supported by which PHYs.

**Table 2-22: PDU type vs. PHYs**

PDU type	PDU name	Channel	Permitted PHY		
			LE 1M	LE 2M	LE coded
0000b	ADV_IND	Primary advertising	x	-	-
0001b	ADV_DIRECT_IND	Primary advertising	x	-	-
0010b	ADV_NONCONN_IND	Primary advertising	x	-	-
0011b	SCAN_REQ	Primary advertising	x	-	-
	AUX_SCAN_REQ	Secondary advertising	x	x	x
0100b	SCAN_RSP	Primary advertising	x	-	-
0101b	CONNECT_IND	Primary advertising	x	-	-
	AUX_CONNECT_REQ	Secondary advertising	x	x	x
0110b	ADV_SCAN_IND	Primary advertising	x	-	-
0111b	ADV_EXT_IND	Primary advertising	x	-	x
	AUX_ADV_IND	Secondary advertising	x	x	x
	AUX_SCAN_RSP	Secondary advertising	x	x	x
	AUX_SYNC_IND	Secondary advertising	x	x	x
	AUX_CHAIN_IND	Secondary advertising	x	x	x
1000b	AUX_CONNECT_RSP	Secondary advertising	x	x	x
Others	Reserved for future use				
x marks supported PHYs					

- The **ChSel**, **TxAdd** and **RxAdd** fields contain information specific to the PDU type. If the ChSel, TxAdd or RxAdd fields are not defined as used in a given PDU then they are considered Reserved for Future Use.
- The **Length** field indicates the payload field length in octets.

### Payload

The advertising channel PDU types can be divided into the following three groups.



**Table 2-23: Advertising channel PDU types**

Advertising PDUs	ADV_IND, ADV_DIRECT_IND, ADV_NONCONN_IND, ADV_SCAN_IND within R&S SMM-K117 also ADV_EXT_IND, AUX_ADV_IND, AUX_SYNC_IND, AUX_CHAIN_IND
Scanning PDUs	SCAN_REQ, SCAN_RSP within R&S SMM-K117 also AUX_SCAN_REQ, AUX_SCAN_RSP
Initiating PDUs	CONNECT_IND within R&S SMM-K117 also AUX_CONNECT_REQ, AUX_CONNECT_RSP

The following parameters are transmitted in the advertising PDU:

- **AdvA, AdvData** for ADV\_IND, ADV\_NONCONN\_IND and ADV\_SCAN\_IND
- **AdvA, TargetA** (formerly InitA) for ADV\_DIRECT\_IND
- **Extended header length, AdvMode, extended header, AdvData** for ADV\_EXT\_IND, AUX\_ADV\_IND, AUX\_SYNC\_IND and AUX\_CHAN\_IND  
**Extended header** contains
  - **AdvA, TargetA, ADI, AuxPtr, Sync Info, Tx power, ACAD,** and **AdvData** fields

The following parameters are transmitted in the scanning PDU:

- **ScanA, AdvA** for SCAN\_REQ  
Within R&S SMM-K117 also for AUX\_SCAN\_REQ
- **AdvA, ScanRspData** for SCAN\_RSP
- **Extended header length, AdvMode, extended header, AdvData** for AUX\_SCAN\_RSP  
**Extended header** contains
  - **AdvA, TargetA, ADI, AuxPtr, Sync Info, Tx power, ACAD,** and **AdvData** fields

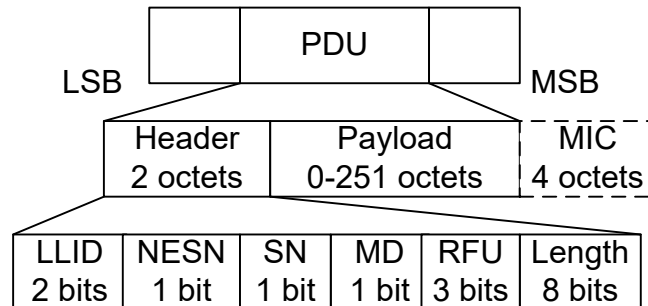
The following parameters are transmitted in the initiating PDU:

- **InitA, AdvA, LLData** for CONNECT\_IND  
Within R&S SMM-K117 also for AUX\_CONNECT\_REQ  
**LLData** contains
  - **AA, CRCinit, WinSize, WinOffset, Interval, Latency, Timeout, ChM, Hop,** and **SCA** fields
- **Extended header length, AdvMode, extended header, AdvData** for AUX\_CONNECT\_RSP  
**Extended header** contains
  - **AdvA, TargetA, ADI, AuxPtr, Sync Info, Tx power, ACAD,** and **AdvData** fields

For more details, refer to in the section 2.3 Advertising Channel PDU of core specification for Bluetooth wireless technology, volume 6, part B.

### 2.3.3.2 Data channel packet structure

#### Header



The 16-bit header field consists of five fields:

- The **LLID** field of the header specifies the payload format, refer to "[Payload](#)" on page 26.
- The **NESN** bit indicates a nextExpectedSeqNum used by the peer to acknowledge the last PDU sent, or to request resending.
- The **SN** bit indicates a transmitSeqNum to identify packets sent by the link layer.
- The **MD** bit indicates, whether the device has more data to send.
- The **Length** field indicates the length of the payload and MIC if included.

#### Payload

- An **LL data PDU** is used to send L2CAP data. The LLID field is set to either 01b or 10b.
  - For the LLID field set to 01b, the LL data PDU is a continuation fragment of an L2CAP message, or an empty PDU. The LL of the Central sends an empty PDU to the Peripheral to allow the Peripheral to respond with any data channel PDU, including an empty PDU.
  - For the LLID field set to 10b, the LL data PDU is a start of an L2CAP message or a complete L2CAP message with no fragmentation.
- An **LL control PDU** is used to control the LL connection. The payload consists of Opcode and CtrData fields. All LL control PDUs have a fixed length, depending on the Opcode. The Opcode field identifies different types of LL Opcode PDU, see [Table 2-20](#).

For more details, refer to in the section 2.4 Data Channel PDU of core specification for Bluetooth wireless technology, volume 6, part B.

### 2.3.4 Modulation scheme

The modulation is Gaussian frequency shift keying (GFSK) with a bandwidth bit period product  $BT = 0.5$ . The modulation index has to be between 0.45 and 0.55. The mandatory modulation scheme is 1 Msymbol/s modulation. It uses a shaped, binary FM to minimize transceiver complexity.

Option R&S SMM-K60 supports LE uncoded 1 Msymbol/s (LE 1M) physical layer (PHY).

Option R&S SMM-K117 supports LE coded 1 Msymbol/s PHY and optional modulation scheme LE uncoded 2 Msymbol/s (LE 2M) PHY.

### 2.3.5 Direction finding

Since Bluetooth version 5.1, a Bluetooth LE device can transmit its direction information to a Bluetooth receiver. The information is transmitted in direction finding enabled packets in the LE uncoded PHY. In combination with location information sent on profile-level, the Bluetooth LE receiver can calculate its position.

#### Angle of Arrival (AoA) method

A Bluetooth LE transmitter sends direction finding enabled packets using a single antenna. A receiving Bluetooth LE peer device consists of an antenna array linked to an RF switch which forwards the combined antennae signal to a Bluetooth LE receiver.

The peer device switches its antennae while receiving parts of the packets and capturing I/Q samples. The I/Q samples are used to calculate the phase difference of the radio signal received by different antennae of the array. For an array of two antennae with distance  $d$ , frequency  $f$  of the radio signal and speed of light  $c$ , the phase difference  $\psi$  calculates as follows:

$$\psi = 2\pi d * \cos(\Theta) * f / c$$

The angle of arrival  $\Theta$  is calculated as follows:

$$\Theta = \arccos((\psi * c) / (2\pi d * f))$$

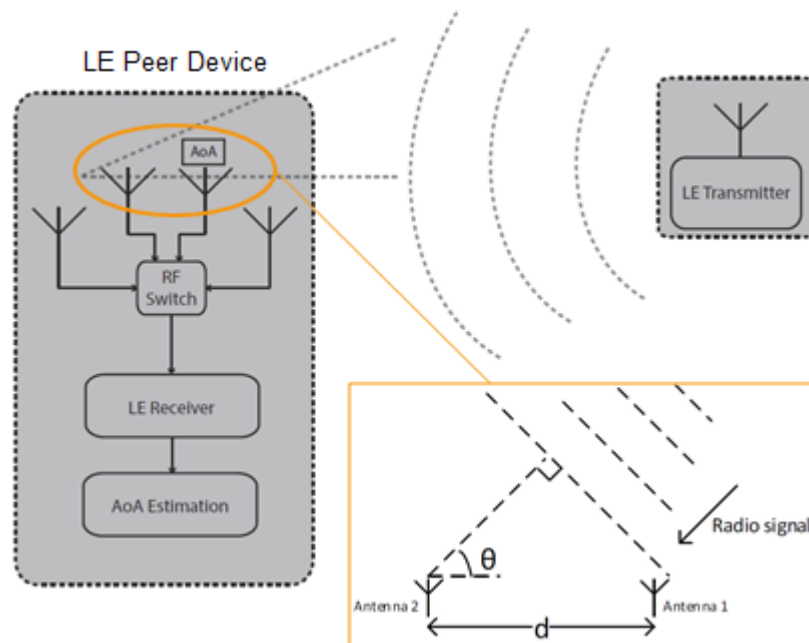


Figure 2-10: Angle of Arrival method

### Angle of Departure (AoD) method

A Bluetooth LE transmitter sends direction finding enabled packets using an antenna array. A receiving Bluetooth LE device, consisting of a single antenna, captures I/Q samples and the geometry of the antenna array from profile-level information.

For an array with two antennae with distance  $d$ , frequency  $f$  of the radio signal and speed of light  $c$ , the phase difference  $\psi$  calculates as follows:

$$\psi = 2\pi d * \cos(\Theta) * f / c$$

The angle of departure  $\Theta$  is calculated as follows:

$$\Theta = \arccos((\psi * c) / (2\pi d * f))$$

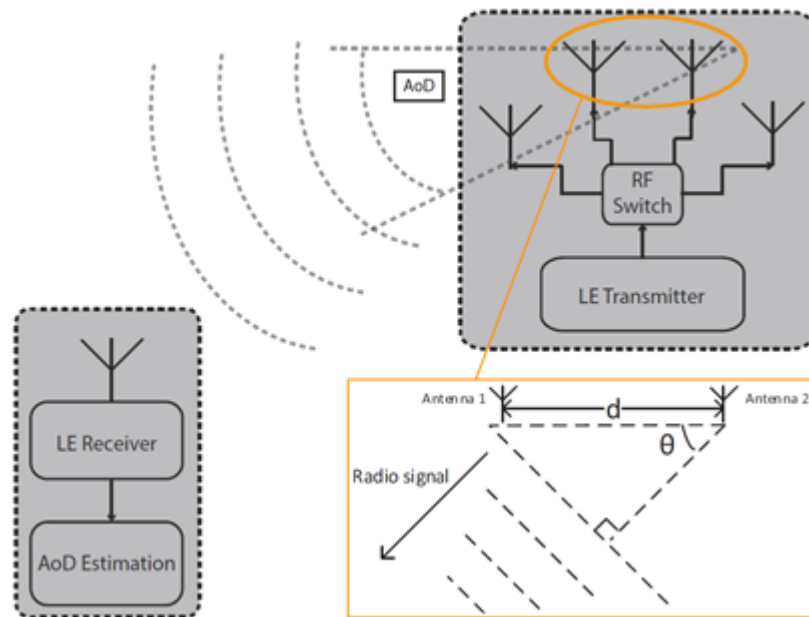


Figure 2-11: Angle of Departure method



The geometry of the antenna array is information that is shared between Bluetooth LE transmitter and receiver on a profile-level. The antenna switching pattern and the method of angle estimation is specified by Constant Tone Extension.

For more information, refer to section 8 Direction Finding Using Bluetooth Low Energy of core specification for Bluetooth wireless technology, volume 1, part A.

### Constant tone extension

To transmit direction finding information in packets in the Bluetooth LE Uncoded PHYs, the link layer packet format is extended by an optional field Constant Tone Extension (CTE) as illustrated in Figure 2-8. The field has a length between 16  $\mu$ s and 160  $\mu$ s and consists of a constantly modulated series of unwhitened 1s. This modulation results in a CW tone shifted by 250 kHz (LE1M) or 500 kHz (LE2M) from the LE channel center frequency.

The presence, type and length of CTE is specified in the CTEInfo field available for ADV\_SYNC\_IND and ADV\_CHAIN\_IND PDUs.

CTEInfo (8 bit)		
CTETime	RFU	CTEType

Figure 2-12: CTEInfo field

The parts of the CTEInfo field are described in the table below. CTEType specifies, if AoA or AoD method is used for direction finding.

CTEInfo field	Length	Value	Description
CTETime	5 bit	2 to 20	CTE length = $8 \mu\text{s} * \text{Value}$ Other values are reserved for future use.
RFU	1 bit	1 to 2	Reserved for future use
CTEType	2 bit	0	AoA Constant tone extension
		1	AoD Constant tone extension with $1 \mu\text{s}$ slots
		2	AoD Constant tone extension with $2 \mu\text{s}$ slots
		3	Reserved for future use

If Bluetooth LE devices support AoA/AoD CTE, the antennae within the array follow a switching pattern specified by the Host. After a guard and reference period, time slots of  $1 \mu\text{s}$  or  $2 \mu\text{s}$  provide periods for antenna switching and I/Q sampling.

The following figure illustrates the CTE structure for AoA method. On the transmitting side, there is no antenna switching. On the receiving side, antenna switching and I/Q sampling alternate in the time slots after the guard and reference period.

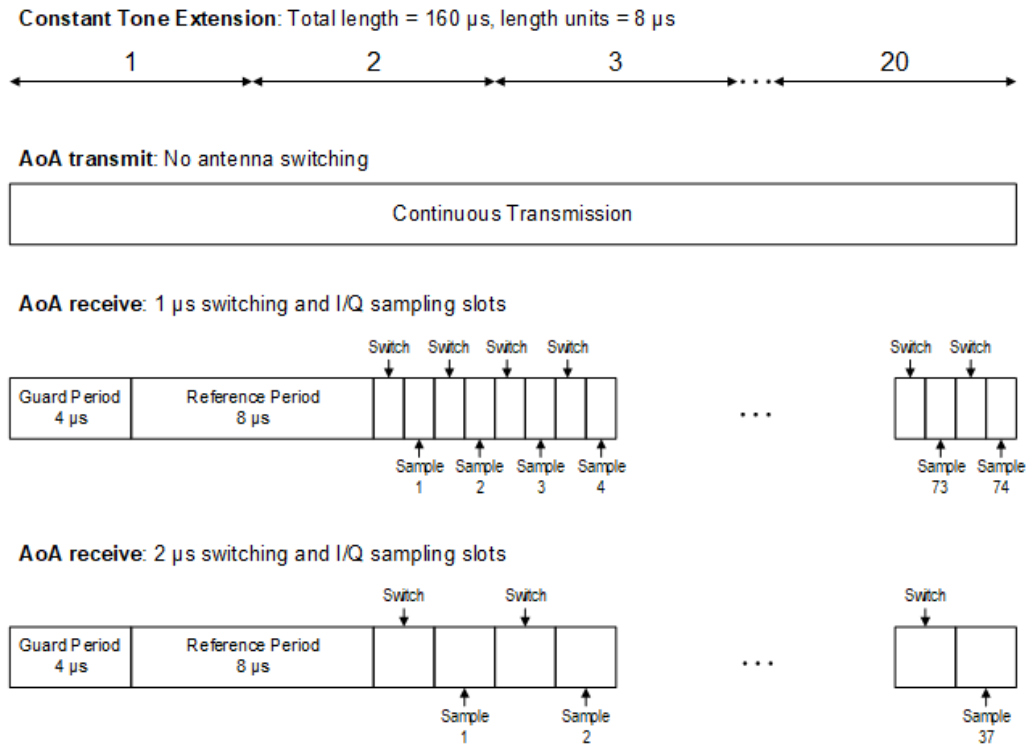
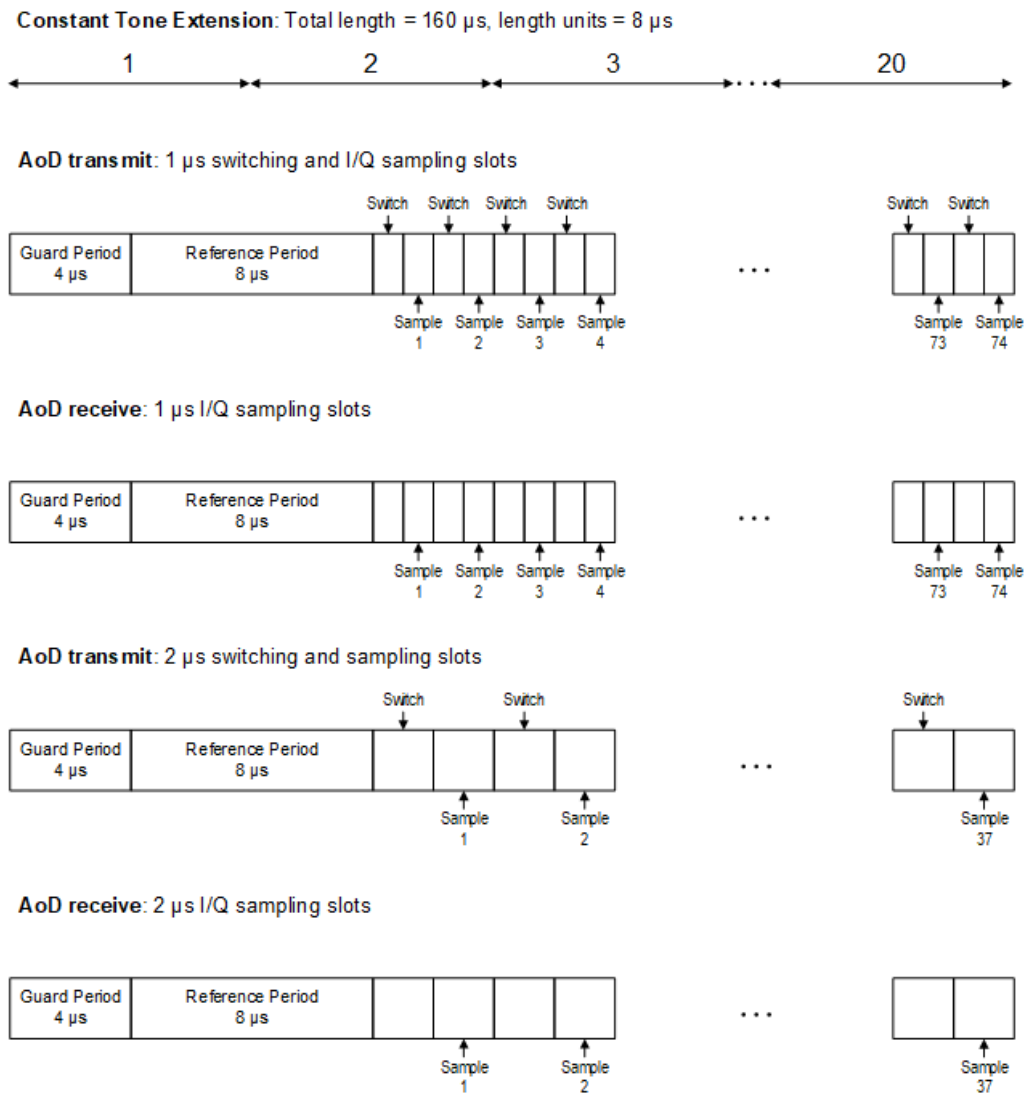


Figure 2-13: CTE structure for AoA method

The following figure illustrates the CTE structure for AoD method. On the transmitting side, antenna switching and I/Q sampling alternate in the time slots after the guard and reference period. On the receiving side, I/Q sampling only is performed in every second time slot after the guard and reference period.



**Figure 2-14: CTE structure for AoD method**

For more information, refer to section 2.5 Constant Tone Extension and IQ Sampling of core specification for Bluetooth wireless technology, volume 6, part B.

## 3 Bluetooth configuration and settings

Access:

- ▶ Select "Baseband" > "Bluetooth".

The remote commands required to define these settings are described in [Chapter 5, "Remote-control commands"](#), on page 106.

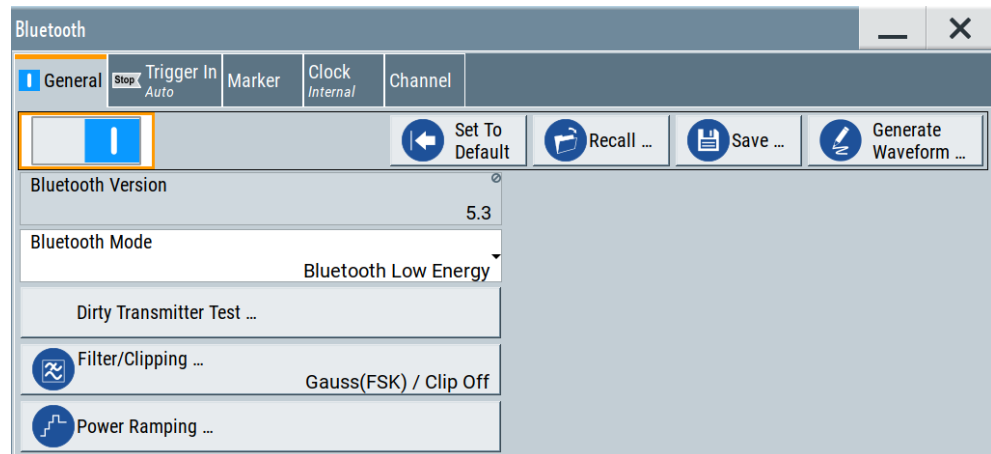
**Settings:**

- [General settings](#).....32
- [Dirty transmitter test settings](#).....35
- [Channel settings - BR/EDR](#).....40
- [Packet configuration - BR/EDR](#).....41
- [Channel settings - LE](#).....49
- [Event / Frame configuration - LE](#).....54
- [Packet configuration - LE](#).....62
- [Test packet configuration - LE](#).....88

### 3.1 General settings

Access:

- ▶ Select "Baseband" > "Bluetooth".



The tab provides general settings to call default settings and the "Save/Recall" settings. Also it provides Bluetooth version information, Bluetooth mode settings and access to further settings.



**Settings:**

State.....	33
Set To Default.....	33
Save/Recall.....	34
Generate Waveform.....	34
Bluetooth Version.....	34
Bluetooth Mode.....	34
Transport Mode.....	34
Dirty Transmitter Test.....	35
Filter/Clipping.....	35
Power Ramping.....	35

**State**

Activates the standard and deactivates all the other digital standards and digital modulation modes in the same path.

Remote command:

[ :SOURce<hw> ] :BB:BT0oth:STATe on page 110

**Set To Default**

Calls the default settings. The values of the main parameters are listed in the following table.

Parameter	Value
State	Not affected by "Set to default"
Bluetooth version	R&S SMM-K60: 4.2 R&S SMM-K117: 5.4
Bluetooth mode	Basic Rate + EDR
Transport mode	ACL(Asynchronous)+EDR
Packet type	DH1
Sequence length	1 Frame
Slot timing	Tx Test Mode
Data whitening	Off
Dirty transmitter test	Off
Filter	Gauss(FSK)
Clipping state	Off
Power ramping function	Cosine
Trigger mode	Auto
Marker mode	Restart
Clock source	Internal

Remote command:

[ :SOURce<hw> ] :BB:BT0oth:PRESet on page 109

**Save/Recall**

Accesses the "Save/Recall" dialog, that is the standard instrument function for saving and recalling the complete dialog-related settings in a file. The provided navigation possibilities in the dialog are self-explanatory.

The settings are saved in a file with predefined extension. You can define the filename and the directory, in that you want to save the file.

See also, chapter "File and Data Management" in the R&S SMM100A user manual.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:SETTing:CATalog` on page 109

`[ :SOURce<hw> ] :BB:BT0oth:SETTing:LOAD` on page 110

`[ :SOURce<hw> ] :BB:BT0oth:SETTing:STORe` on page 110

`[ :SOURce<hw> ] :BB:BT0oth:SETTing:DELeTe` on page 110

**Generate Waveform**

With enabled signal generation, triggers the instrument to save the current settings of an arbitrary waveform signal in a waveform file with predefined extension `*.wv`. You can define the filename and the directory, in that you want to save the file.

Using the ARB modulation source, you can play back waveform files and/or process the file to generate multi-carrier or multi-segment signals.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:WAVeform:CREate` on page 112

**Bluetooth Version**

Displays the version of the standard that the firmware supports.

The displayed version for Bluetooth wireless technology depends on installed options.

For example, "Bluetooth Version" > "5.4" in accordance with Bluetooth core specification v5.4, requires R&S SMM-K117.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:VERSion?` on page 111

**Bluetooth Mode**

Determines the Bluetooth mode.

"Basic Rate +EDR"

Selects the standard Bluetooth mode (BR+EDR).

Specific settings of the basic mode are described in [Chapter 3.3, "Channel settings - BR/EDR"](#), on page 40.

"Bluetooth Low Energy"

Selects the Bluetooth LE mode. Specific settings of this mode are described in [Chapter 3.5, "Channel settings - LE"](#), on page 49.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:BMODE` on page 109

**Transport Mode**

Requires "Bluetooth Mode > Basic Rate + EDR".

Selects the transport mode.

- "ACL+EDR" The transport mode selected is used for a point-to-multipoint link establishment between the Central and all the Peripherals participating on the piconet.
- "SCO" The transport mode selected is used for a point-to-point link establishment between a Central and a single Peripheral in the piconet.
- "eSCO+EDR" The transport mode selected is used for a symmetric or asymmetric point-to-point link establishment between a Central and a specific Peripheral.

Remote command:

[ :SOURCE<hw> ] :BB:BT0oth:TMODe on page 111

### Dirty Transmitter Test

Accesses the [Dirty transmitter test settings](#) dialog, see [page 35](#).

### Filter/Clipping

Accesses the dialog for setting baseband filtering, the modulation settings and clipping, see [Chapter 4.1, "Filter/clipping settings"](#), on page 90.

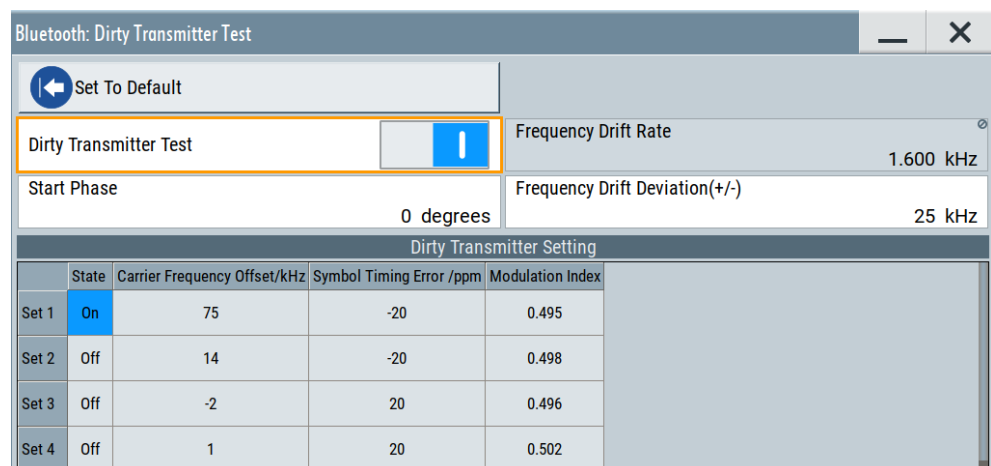
### Power Ramping

Accesses the dialog for setting power ramping, see [Chapter 4.2, "Power ramping settings"](#), on page 96.

## 3.2 Dirty transmitter test settings

Access:

1. Select "Bluetooth" > "General".
2. Select "Dirty Transmitter Test".



The dialog provides setting to configure dirty transmitter test settings. These settings contain parameters that you can change for the central device signal. It is used to test the connection under 'dirty transmitter' conditions, and to define the influence on the receiver quality (bit error rate tests).

The following tables list dirty transmitter parameters according to the Bluetooth test specification.  $f_{\text{offset}}$  is the frequency offset,  $\Delta t_{\text{error}}$  is the symbol timing error and  $h$  is the modulation index.

**Table 3-1: Dirty transmitter test parameters for BR**

Set	$f_{\text{offset}}$	$\Delta t_{\text{error}}/\text{ppm}$	$h$
1	75	-20	0.28
2	14	-20	0.30
3	-2	20	0.29
4	1	20	0.32
5	39	20	0.33
6	0	-20	0.34
7	-42	-20	0.29
8	74	-20	0.31
9	-19	-20	0.28
10	-75	20	0.35

**Table 3-2: Dirty transmitter test parameters for EDR**

Set	$f_{\text{offset}}$	$\Delta t_{\text{error}}/\text{ppm}$
1	0	0
2	65	20
3	-65	-20

**Table 3-3: Dirty transmitter test parameters for LE**

Set	$f_{\text{offset}}$	$\Delta t_{\text{error}}/\text{ppm}$	$h_{\text{standard}}$	$h_{\text{stable}}^*)$
1	100	-50	0.45	0.495
2	19	-50	0.48	0.498
3	-3	50	0.46	0.496
4	1	50	0.52	0.502
5	52	50	0.53	0.503
6	0	-50	0.54	0.504
7	-56	-50	0.47	0.497
8	97	-50	0.50	0.500

Set	$f_{\text{offset}}$	$\Delta t_{\text{error}}/\text{ppm}$	$h_{\text{standard}}$	$h_{\text{stable}}^*)$
9	-25	-50	0.45	0.495
10	-100	50	0.55	0.505

\*)  $h_{\text{stable}}$  requires R&S SMM-K117.

### Settings:

Set To Default.....	37
Dirty Transmitter Test.....	37
Start Phase.....	37
Modulation Index Mode.....	38
Frequency Drift Rate.....	38
Frequency Drift Deviation (+/-).....	38
Number of Packets per Set.....	38
Dirty Transmitter Setting.....	38
L State.....	38
L Carrier Frequency Offset kHz.....	39
L Symbol Timing Error.....	39
L Modulation Index.....	39

### Set To Default

Calls the default settings for the dirty transmitter test. Default settings are according to the specification for Bluetooth wireless technology. The setting corresponds the selected Bluetooth mode.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:STDefault` on page 115

### Dirty Transmitter Test

Activates or deactivates the dirty transmitter test.

The setting is available for the following packet types:

- **BR:** DH1, DH3, DH5
- **EDR:** 2-DH1, 2-DH3, 2-DH5, 3-DH1, 3-DH3, 3-DH5, 2-EV3, 2-EV5, 3-EV3, 3-EV5
- **LE:** See the tables [Table 3-4](#) and [Table 3-5](#).

For basic rate packets, each enabled set of parameters in the "Dirty Transmitter Setting" is used for a duration of 20 ms. After 20 ms, the following enabled set is used, continuing with the first enabled set after the sequence is completed.

For EDR packets, the parameter sets apply for 20 packets each.

For LE, each enabled set of parameters in the "Dirty Transmitter Setting" is used. After the specified [Number of Packets per Set](#) (specification defines 50 packets) is transmitted, a following enabled set is used. After the sequence is completed, the transmission continues with the first enabled set.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:DTTState` on page 113

### Start Phase

Enters a start phase.

The start phase of the sine wave used to drift the modulated Bluetooth signal around center frequency + carrier frequency offset is set here.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:SPHase` on page 115

#### Modulation Index Mode

Requires option R&S SMM-K117 and "Bluetooth Mode" > "Bluetooth Low Energy".

Specifies the mode of the modulation index h that is standard or stable.

"Standard"      Standard modulation index with range  $h_{\text{standard}} = 0.450$  to  $h = 0.550$

"Stable"         Stable modulation index with range  $h_{\text{stable}} = 0.495$  to  $h = 0.505$

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:MIMode` on page 114

#### Frequency Drift Rate

Sets the frequency drift rate.

A sine wave is used to drift the modulated Bluetooth signal around center frequency + carrier frequency offset with the set frequency drift rate.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:FDRate` on page 114

#### Frequency Drift Deviation (+/-)

Enters a frequency drift deviation.

A sine wave is used to drift the modulated Bluetooth signal around center frequency + carrier frequency offset. The maximum deviation reached during the drift equals the set frequency drift deviation.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:FDDeviation` on page 113

#### Number of Packets per Set

For "Bluetooth Mode = Bluetooth Low Energy", specifies the number of test packets to be transmitted per enabled dirty transmitter set.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:DTTest:NPPSet` on page 114

#### Dirty Transmitter Setting

Indicates the dirty transmitter parameters according to the Bluetooth BR test specification.

#### State ← Dirty Transmitter Setting

Activates or deactivates the corresponding parameter set.

If deactivated, the parameters are skipped in the sequence, and the next active set is used.

Remote commands `...:LONG:SET<ch>:...` are used for BR and LE packets. The instrument provides configuration of up to 10 sets (SET1 to SET10).

Remote commands . . . :SHORT:SET<ch>: . . . are used for EDR packets. The instrument provides configuration of up to 3 sets (SET1 to SET3).

For basic rate packets, each enabled set applies to 20ms of signal. For EDR packets, each enabled set applies to 20 packets.

For LE, each enabled set applies to 50 test packets.

Remote command:

[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TABLE:LONG:SET<ch>:STATE  
on page 117

[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TABLE:SHORT:SET<ch>:STATE  
on page 118

### Carrier Frequency Offset kHz ← Dirty Transmitter Setting

Determines a carrier frequency offset.

The center frequency of the modulated RF carrier is offset by the specified value.

Remote command:

[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TABLE:LONG:SET<ch>:CFOffset  
on page 115

[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TABLE:SHORT:SET<ch>:CFOffset  
on page 117

### Symbol Timing Error ← Dirty Transmitter Setting

Sets the symbol timing error in ppm.

The symbol timing error modifies the symbol clock frequency by the specified value.

Remote command:

[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TABLE:LONG:SET<ch>:STERror  
on page 117

[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TABLE:SHORT:SET<ch>:STERror  
on page 118

### Modulation Index ← Dirty Transmitter Setting

(Only for basic rate packets)

Sets the modulation index.

The modulation index *h* specifies the frequency deviation, defined as:

$$h = \frac{2\Delta f}{f_{symbol}}$$

$f_{symbol}$  is the "symbol rate" and  $\Delta f$  is the "frequency deviation".

According to the Bluetooth specification, the modulation index can vary between 0.28 and 0.35.

Remote command:

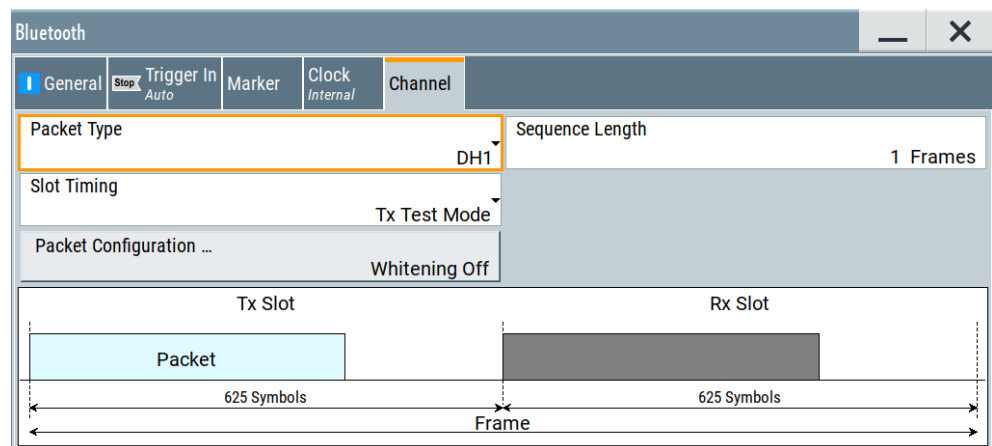
[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TABLE:LONG:SET<ch>:MINDex  
on page 116

### 3.3 Channel settings - BR/EDR

This dialog provides access to the "Bluetooth Basic Rate + EDR" settings. For LE settings, refer to [Chapter 3.5, "Channel settings - LE"](#), on page 49.

Access:

1. Select "Bluetooth > General > Bluetooth Mode > Basic Rate + EDR".
2. Select "Channel".



The dialog contains the parameters to define the packet type and provides access to the packet type configuration dialog. The graphic shows the frame structure of the selected packet type.

#### Settings:

<a href="#">Packet Type</a> .....	40
<a href="#">Sequence Length</a> .....	40
<a href="#">Slot Timing</a> .....	41
<a href="#">Packet Configuration</a> .....	41

#### Packet Type

Selects the packet type.

The available packets depend on the selected [Transport Mode](#).

All packet types as defined in the Bluetooth specification are supported. For an overview, see [Chapter 2.2.1, "Bluetooth packet types for BR/EDR"](#), on page 13.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:PTYPe` on page 119

#### Sequence Length

Selects the sequence length in frames of the generated signal. The signal repeats after the specified number of frames.

Remote command:

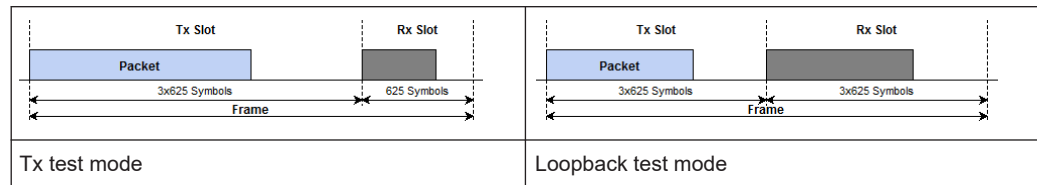
`[ :SOURce<hw> ] :BB:BT0oth:SLENgth` on page 119



### Slot Timing

Selects the timing mode for the Rx slot.

The graphic below shows the frame structure of the selected **Packet Type** and slot timing.



A transmitted packet has a duration of  $N \times 625 \mu\text{s}$  where  $N$  is an odd integer larger than 0.  $N$  depends on the type of the transmitted packet. In "Tx Test" mode,  $N = 1$  for Rx slots.

#### "Tx Test Mode"

The transmitted Rx package takes 625 symbols, regardless of the selected packet type.

#### "Loopback Test Mode"

Extends the Rx slot time according to the selected packet type.

For example, the Rx slot of **Packet Type > DH3** takes  $3 \times 625$  symbols.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:STIMing` on page 120

### Packet Configuration

Access the "Packet Configuration" dialog, see [Chapter 3.4, "Packet configuration - BR/EDR"](#), on page 41.

The current data source for packet and the data whitening state are displayed next to the button.

Remote command:

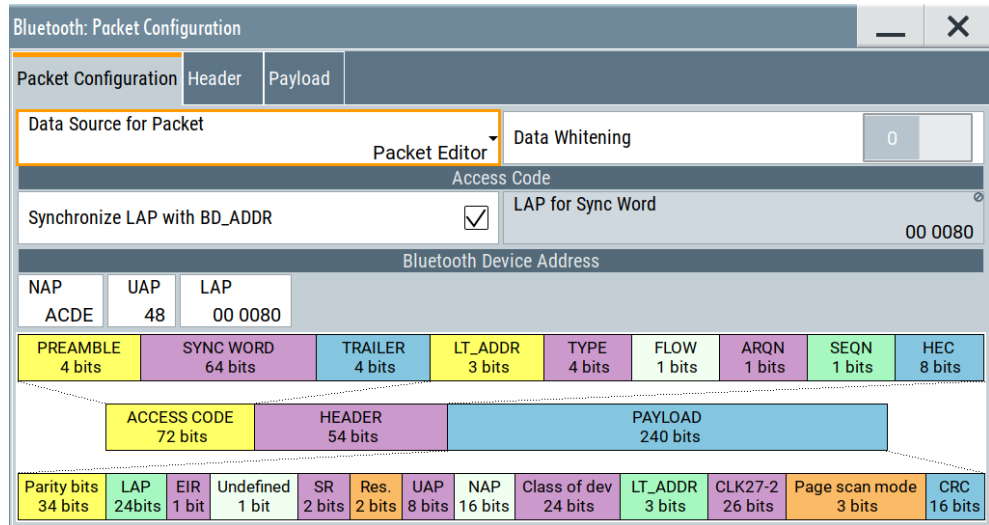
n.a.

## 3.4 Packet configuration - BR/EDR

Access:

1. Select "Bluetooth > General > Bluetooth Mode > Basic Rate + EDR".

2. Select "Bluetooth > Channel > Packet Configuration".



The dialog contains the parameters for configuring the packet type. Available settings vary according to the selected **Packet Type** and data source.

**Settings:**

- Packet Configuration.....43
  - Data Source for Packet.....43
  - Data Whitening.....43
  - Synchronize LAP with BD\_ADDR.....43
  - LAP for Sync Word.....43
  - Bluetooth Device Address (BD\_ADDR).....43
- Header.....44
  - Logical Transport Address.....44
  - Flow Control.....44
  - Acknowledgment.....44
  - SEQN Start Value.....45
- Payload.....45
  - Data Source.....45
  - Data Length.....46
  - EIR packet follows.....46
  - Flow Control.....46
  - Scan Repetition Mode.....46
  - Class of Device.....47
- DV Payload.....47
  - Data Source (Voice Field).....47
  - Data Source.....48
  - Data Length.....48
  - Flow Control.....48
- Data.....49
  - Packet Length.....49

**Packet Configuration**

In this section, specify general Bluetooth BR/EDR packet properties.

**Data Source for Packet ← Packet Configuration**

The data sent for each packet can be comfortably edited with the packet editor, or filled with a predefined ALL data sequence.

"Packet Editor" Enables the edit mode to configure the packet fields individually.

"All Data" Fills the generated packets with the selected data source. This mode is useful if you need to load predefined data contents from a data list file or the data contents of the packet are not of interest.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:PCONfiguration:DSFPacket` on page 124

**Data Whitening ← Packet Configuration**

Activates the data whitening.

Evenly distributed white noise is ideal for the transmission, and real data can be forced to look similar to white noise with different methods called "Data Whitening".

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:PCONfiguration:DWhitening` on page 125

**Synchronize LAP with BD\_ADDR ← Packet Configuration**

(Available for FHS packets)

Activates synchronization of the [LAP for Sync Word](#) and the [Bluetooth Device Address > LAP](#).

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:PCONfiguration:SLAP` on page 128

**LAP for Sync Word ← Packet Configuration**

(Available for FHS packets)

Sets the 24 bits lower address part (LAP) in the 64 bits sync word separately, if "Synchronize LAP with BD\_ADDR > OFF".

The LAP is obtained automatically from the Bluetooth device address "BD\_ADDR > LAP", if "Synchronize LAP with BD\_ADDR > ON".

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:PCONfiguration:LFSWord` on page 126

**Bluetooth Device Address (BD\_ADDR) ← Packet Configuration**

Enters the Bluetooth device address. Each Bluetooth device has allocated a unique 48-bit Bluetooth device address (BD\_ADDR).

The BD\_ADDR can take any values except the 64 reserved LAP values: 0x9E8B00 – 0x9E8B3F.

"NAP" Selects non-significant address part.  
The length of NAP is 16 bits or 4 hexadecimal figures.

"UAP" Selects upper address part.  
The length of UAP is 8 bits or two hexadecimal figures.

"LAP" Selects lower address part.  
The length of LAP is 24 bits or 6 hexadecimal figures.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:BDANap](#) on page 121

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:BDAUap](#) on page 122

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:BDALap](#) on page 121

### Header

Access:

Select "Bluetooth > General > Bluetooth Mode > Basic Rate + EDR > Packet Configuration > Header".

Packet Configuration	Header	Payload
LT Address	0	Flow Control
Acknowledgment	ACK	SEQN Start Value
		GO
		1

Provides header settings.

#### Logical Transport Address ← Header

Available for all packet types except ID packet.

Enters the logical transport address for the header.

Each Peripheral, that is active in a piconet, is assigned a primary logical transport address (LT\_ADDR). The all-zero LT\_ADDR is reserved for broadcast messages.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:LTAddress](#) on page 126

#### Flow Control ← Header

Available for all packet types except ID packet.

Sets the FLOW bit in the header. This bit indicates start or stop of transmission of packets over the ACL logical transport.

"Go" Allows the other devices to transmit new data.

"Stop" Stops the other devices from transmitting data temporarily.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:HFCControl](#) on page 126

#### Acknowledgment ← Header

Available for all packet types except ID packet.

Sets the ARQN bit of the packet header.

"NAK" Request to retransmit the previous payload.

"ACK" Previous payload has been received successfully.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:ACKnowledgement](#) on page 120

**SEQN Start Value ← Header**

Available for all packet types except ID packet.

Sets the start value of the header SEQN bit.

The SEQN bit is present in the header to filter out retransmissions in the destination. The signal generator is altering this bit automatically on consecutive frames, if a sequence length of at least two frames is set.

"0"                    The SEQN bit starts with 0.

"1"                    The SEQN bit starts with 1.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:SNSValue](#) on page 128

**Payload**

Access:

Select "Bluetooth > General > Bluetooth Mode > Basic Rate + EDR > Packet Configuration > Payload".

Packet Configuration	Header	Payload
Data Source		PN 9
Data Length	17 bytes	Flow Control GO

Provides payload settings.

**Data Source ← Payload**

(Available for all packet types except ID, POLL, NULL and FHS packets)

Selects the data source used for the payload.

The following standard data sources are available:

- "All 0, All 1"  
An internally generated sequence containing 0 data or 1 data.
- "PNxx"  
An internally generated pseudo-random noise sequence.
- "Pattern"  
An internally generated sequence according to a bit pattern.  
Use the "Pattern" box to define the bit pattern.
- "Data List/Select DList"  
A binary data from a data list, internally or externally generated.  
Select "Select DList" to access the standard "Select List" dialog.
  - Select the "Select Data List > navigate to the list file \*.dm\_iqd > Select" to select an existing data list.
  - Use the "New" and "Edit" functions to create internally new data list or to edit an existing one.
  - Use the standard "File Manager" function to transfer external data lists to the instrument.

See also:

- Section "Modulation Data" in the R&S SMM100A user manual.
- Section "File and Data Management" in the R&S SMM100A user manual.

- Section "Data List Editor" in the R&S SMM100A user manual

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:PCONfiguration:DATA` on page 122

`[ :SOURCE<hw> ] :BB:BT00th:PCONfiguration:DATA:DPATtern` on page 123

`[ :SOURCE<hw> ] :BB:BT00th:PCONfiguration:DATA:DSElection` on page 123

#### **Data Length ← Payload**

(Available for all packet types except ID, POLL, NULL and FHS packets)

Enters the payload data length in bytes.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:PCONfiguration:DLENgth` on page 124

#### **EIR packet follows ← Payload**

Available for FHS packets.

Indicates that an extended inquiry response packet can follow.

"Yes" Indicates that an EIR packet follows.

"No" Indicates that EIR does not follow.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:PCONfiguration:EIRPacketfollows`  
on page 125

#### **Flow Control ← Payload**

(Available for all packets types except ID, POLL, NULL, FHS, HV1, HV2, HV3, EV3, EV4, EV5, 2-EV3, 2-EV5, 3-EV3, 3-EV5 packets.)

Sets the FLOW bit in the payload (flow control per logical link)

"Go" Indicates start of transmission of ACL packets after a new connection has been established.

"Stop" Indicates stop of transmission of ACL packets before an additional amount of payload data is sent.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:PCONfiguration:PFControl` on page 127

#### **Scan Repetition Mode ← Payload**

Available for FHS packets.

The 2-bit scan repetition field indicates the interval between two consecutive page scan windows, determines the behavior of the paging device.

"R0" The scan interval is equal to the scan window  $T_{w\_page\_scan}$  (continuous scan) and maximal 1.28s.

"R1" The scan interval is maximal 1.28s.

"R2" The scan interval is maximal 2.56s.

Remote command:

`[ :SOURCE<hw> ] :BB:BT00th:PCONfiguration:SRMode` on page 128

**Class of Device ← Payload**

Available for FHS packets.

A parameter received during the device discovery procedure, indicates the type of device and which types of service that are supported.

Remote command:

[ :SOURce<hw> ] :BB:BT0oth:PCONfiguration:CODevice on page 122

**DV Payload**

Access:

Select "Bluetooth > Transport Mode = SCO > Channel > Packet Type = DV > Packet Configuration > Data Source for Packet = Packet Editor > DV Payload".

Packet Configuration	Header	DV Payload	
		Voice Field	
Data Source		PN 9	
		Data Field	
Data Source		Data List	Select Data List ... BR_Data_Test
Data Length		9 bytes	Flow Control GO

Provides DV payload settings.

**Data Source (Voice Field) ← DV Payload**

(Available for DV packets)

Selects the data source for the voice field.

The following standard data sources are available:

- "All 0, All 1"  
An internally generated sequence containing 0 data or 1 data.
- "PNxx"  
An internally generated pseudo-random noise sequence.
- "Pattern"  
An internally generated sequence according to a bit pattern.  
Use the "Pattern" box to define the bit pattern.
- "Data List/Select DList"  
A binary data from a data list, internally or externally generated.  
Select "Select DList" to access the standard "Select List" dialog.
  - Select the "Select Data List > navigate to the list file \*.dm\_iqd > Select" to select an existing data list.
  - Use the "New" and "Edit" functions to create internally new data list or to edit an existing one.
  - Use the standard "File Manager" function to transfer external data lists to the instrument.

See also:

- Section "Modulation Data" in the R&S SMM100A user manual.
- Section "File and Data Management" in the R&S SMM100A user manual.
- Section "Data List Editor" in the R&S SMM100A user manual

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:VDATA](#) on page 129

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:DATA:VDPattern](#) on page 123

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:DATA:VDSElection](#)  
on page 124

### Data Source ← DV Payload

(Available for all packet types except ID, POLL, NULL and FHS packets)

Selects the data source used for the payload.

The following standard data sources are available:

- "All 0, All 1"  
An internally generated sequence containing 0 data or 1 data.
- "PNxx"  
An internally generated pseudo-random noise sequence.
- "Pattern"  
An internally generated sequence according to a bit pattern.  
Use the "Pattern" box to define the bit pattern.
- "Data List/Select DList"  
A binary data from a data list, internally or externally generated.  
Select "Select DList" to access the standard "Select List" dialog.
  - Select the "Select Data List > navigate to the list file \*.dm\_iqd > Select" to select an existing data list.
  - Use the "New" and "Edit" functions to create internally new data list or to edit an existing one.
  - Use the standard "File Manager" function to transfer external data lists to the instrument.

See also:

- Section "Modulation Data" in the R&S SMM100A user manual.
- Section "File and Data Management" in the R&S SMM100A user manual.
- Section "Data List Editor" in the R&S SMM100A user manual

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:DATA](#) on page 122

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:DATA:DPATtern](#) on page 123

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:DATA:DSElection](#) on page 123

### Data Length ← DV Payload

(Available for all packet types except ID, POLL, NULL and FHS packets)

Enters the payload data length in bytes.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:DLENgth](#) on page 124

### Flow Control ← DV Payload

(Available for all packets types except ID, POLL, NULL, FHS, HV1, HV2, HV3, EV3, EV4, EV5, 2-EV3, 2-EV5, 3-EV3, 3-EV5 packets.)

Sets the FLOW bit in the payload (flow control per logical link)

"Go"                      Indicates start of transmission of ACL packets after a new connection has been established.



"Stop" Indicates stop of transmission of ACL packets before an additional amount of payload data is sent.

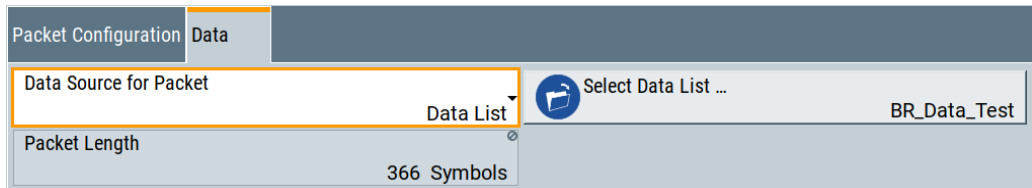
Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:PFControl](#) on page 127

### Data

Access:

Select "Packet Configuration > Data Source for Packet = All Data > Data".



Provides data settings.

### Packet Length ← Data

(Available in "All Data" mode and for all packet types except ID packet.)

Enters the packet length in symbols.

Remote command:

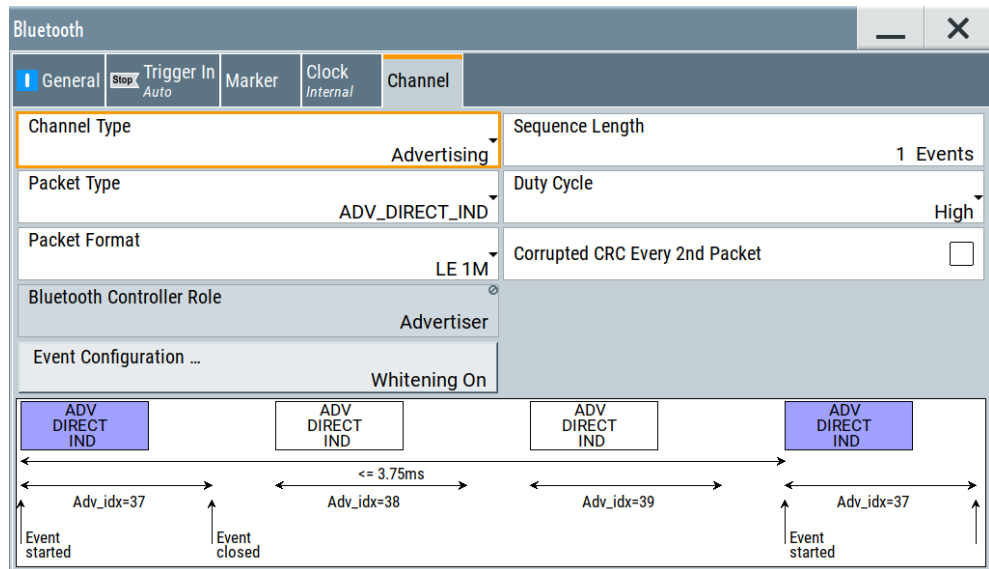
[\[:SOURCE<hw>\]:BB:BT0oth:PCONfiguration:PLENgtH](#) on page 127

## 3.5 Channel settings - LE

Access:

1. Select "Bluetooth" > "General".
2. Select "Bluetooth Mode" > "Bluetooth Low Energy".
3. Select "Channel".

The tab provides settings to configure channel settings and general packet settings.

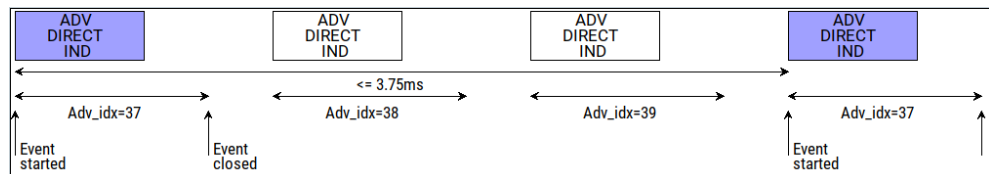


**To display Bluetooth LE event or frame sequences**

Depending on the channel type packet type and packet format, the "Channel" tab displays a chart at the bottom that illustrates sequences of the corresponding events or frames. The following steps provide examples to illustrate default sequences for different channel types.

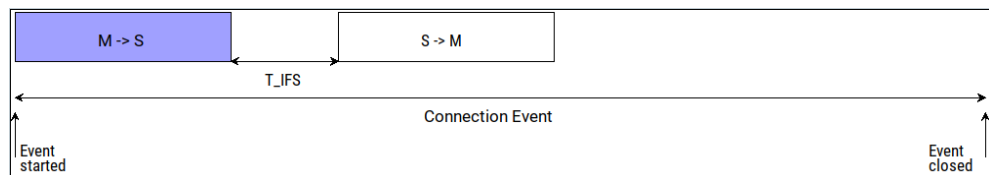
1. To display the sequence for an advertising channel, select the following:
  - a) Select "Channel Type" > "Advertising".
  - b) Select, for example, "Packet Type" > "ADV\_DIRECT\_IND".

The chart displays one sequence of ADV\_DIRECT\_IND packets including markers for event start and event stop.



2. To display the sequence for a data channel, select the following:
  - a) Select "Channel Type" > "Data".
  - b) Select, for example, "Packet Type" > "DATA".

The chart displays one sequence of DATA PDUs including markers for event start and event stop.



**Settings:**

Channel Type.....	51
Packet Type.....	51
Packet Format.....	52
Duty Cycle.....	52
Sequence Length.....	53
Bluetooth Controller Role.....	53
Bluetooth Controller State.....	53
Corrupted CRC Every 2nd Packet.....	53
Payload Type.....	54
Duration.....	54
Modulation Format.....	54
Event Configuration/Frame Configuration.....	54
Test Packet Configuration.....	54

**Channel Type**

Determines the channel type.

- "Advertising"      Selects channel type advertising.
- "Data"                Selects the data channel type.  
Devices in a connected state transmit the data channel packets in connection events with a start point and an interval.

Remote command:

[ :SOURce<hw> ] :BB:BT0oth:CTYPe on page 131

**Packet Type**

Selects the packet type that is the PDU type.

Available packet types depend on the channel type and installed options, see the tables [Table 3-4](#) and [Table 3-5](#).

**Table 3-4: Packet/PDU types and channel types**

Packet/PDU type	Channel type	Option
ADV_IND, ADV_DIRECT_IND, ADV_NONCONN_IND, ADV_SCAN_IND, SCAN_REQ, SCAN_RSP, CONNECT_IND	Advertising	R&S SMM-K60
DATA, CONTROL_DATA	Data	R&S SMM-K60
TEST PACKET	Advertising, data	R&S SMM-K60
CONTINUOUS	Advertising, Data	R&S SMM-K60
ADV_EXT_IND *), AUX_ADV_IND, AUX_CHAIN_IND, AUX_SYNC_IND, AUX_SCAN_REQ, AUX_SCAN_RSP, AUX_CONNECT_REQ, AUX_CONNECT_RSP	Advertising	R&S SMM-K117

\*) PDU type ADV\_EXT\_IND is available for LE 1M PHY and LE Coded PHY. For more information, see data sheet.

CONTROL\_DATA packets/PDUs depend on the controller role and installed options.

**Table 3-5: CONTROL\_DATA PDU and controller role**

CONTROL_DATA PDU	Controller role	Option
LL_CONNECTION_UPDATE_IND, LL_CHANNEL_MAP_IND, LL_ENC_REQ, LL_FEATURE_REQ, LL_PAUSE_ENC_REQ	Central	R&S SMM-K60
LL_ENC_RSP, LL_UNKNOWN_RSP, LL_FEATURE_RSP, LL_PAUSE_ENC_RSP	Peripheral	R&S SMM-K60
LL_TERMINATE_IND, LL_START_ENC_REQ, LL_START_ENC_RSP, LL_VERSION_IND, LL_REJECT_IND	Central or Peripheral	R&S SMM-K60
LL_PHY_UPDATE_IND	Central	R&S SMM-K117
LL_PERIPHERAL_FEAT_REQ, LL_CONNECTION_PARAM_RSP, LL_PHY_RSP, LL_MIN_USED_CHANNELS_IND	Peripheral	R&S SMM-K117
LL_CONNECTION_PARAM_REQ, LL_REJECT_EXT_IND, LL_PING_REQ, LL_PING_RSP, LL_LENGTH_REQ, LL_LENGTH_RSP, LL_PHY_REQ, LL_CTE_REQ, LL_CTE_RSP, LL_PERIODIC_SYNC_IND, LL_CLOCK_ACCURACY_REQ, LL_CLOCK_ACCURACY_RSP	Central or Peripheral	R&S SMM-K117

How to: ["To display Bluetooth LE event or frame sequences"](#) on page 50

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:UPTYPE` on page 132

### Packet Format

Selects the packet format that is the format according to the physical layer (PHY) that supports the PDU type.

- "LE 1M" LE uncoded PHY with 1 Msymbol/s modulation
- "LE 2M" Requires R&S SMM-K117.  
LE uncoded PHY with 2 Msymbol/s modulation
- "LE Coded" Requires R&S SMM-K117.  
LE coded PHY with 1 Msymbol/s modulation

See also [Table 2-22](#).

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:PFORmat` on page 131

### Duty Cycle

Requires R&S SMM-K117 and "Packet Type" > "ADV\_DIRECT\_IND".

Specifies the duty cycle for directed advertising.

- "High" Transmits the "ADV\_DIRECT\_IND" packet expecting an advertising event interval.
- "Low" Transmits the "ADV\_DIRECT\_IND" packet expecting an advertising event interval and advertising event delay.

See also ["Advertising Event Interval"](#) on page 56.

Remote command:

[ :SOURce<hw> ] :BB:BT0oth:DCYClE on page 131

### Sequence Length

Selects the number of frames or events depending on the packet type. The signal repeats after this number.

The packet types SCAN\_REQ, CONNECT\_IND, AUX\_SCAN\_REQ and AUX\_CONNECT\_REQ define the sequence length in frames. All other packets use events.

Remote command:

[ :SOURce<hw> ] :BB:BT0oth:USLength on page 133

### Bluetooth Controller Role

Requires "Channel Type" > "Advertising"/"Data".

Determines the controller role. Depending on the channel type, displays the controller role or you can select it.

Bluetooth controller role	Channel type	PDU type
Advertiser (read-only)	Advertising	ADV_IND, ADV_DIRECT_IND, ADV_NON-CONN_IND, ADV_SCAN_IND, SCAN_RSP, ADV_EXT_IND, AUX_ADV_IND, AUX_CHAIN_IND, AUX_SYNC_IND, AUX_SCAN_RSP, AUX_CONNECT_RSP, TEST PACKET
Scanner (read-only)	Advertising	SCAN_REQ, AUX_SCAN_REQ
Initiator (read-only)	Advertising	CONNECT_IND, AUX_CONNECT_IND
Central	Data	DATA, CONTROL_DATA
Peripheral	Data	DATA, CONTROL_DATA

See also "Payload" on page 24 and Table 3-5.

Remote command:

[ :SOURce<hw> ] :BB:BT0oth:BCRole on page 130

### Bluetooth Controller State

Requires "Packet Type" > "Data"/"CONTROL\_DATA".

Displays the state of the Bluetooth controller.

"Connected" Controller connected using a data channel.

"Disconnected" Controller disconnected.

Remote command:

[ :SOURce<hw> ] :BB:BT0oth:BCText? on page 108

### Corrupted CRC Every 2nd Packet

If enabled, sets the ratio of packets with CRC faults to 50%. 50% of packets are generated with correct CRC. This setting is appropriate for packet error rate (PER) report integrity tests.

Remote command:

[ :SOURce<hw> ] :BB:BT0oth:CCRC:STATe on page 130

**Payload Type**

Requires "Packet Type" > "CONTINUOUS".

Specifies the pattern for continuous packets. The transmitted packets do not contain packet header information.

For supported payload types, refer to ["Payload Type"](#) on page 89.

**Duration**

Requires "Packet Type" > "CONTINUOUS".

Specifies the transmission duration.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:DURation](#) on page 134

**Modulation Format**

Requires "Packet Type" > "CONTINUOUS".

Specifies the physical layer.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:MFORmat](#) on page 133

**Event Configuration/Frame Configuration**

Access the configuration "Event Configuration" dialog, if the sequence length of the packet type is expressed in events, and accordingly, the "Frame Configuration" dialog, if it is expressed in frames, see [Chapter 3.6, "Event / Frame configuration - LE"](#), on page 54.

Also, the button displays the data whitening state, see ["Data Whitening"](#) on page 63.

**Test Packet Configuration**

Requires "Packet Type" > "TEST PACKET".

Accesses the "Test Packet Configuration" dialog, see [Chapter 3.8, "Test packet configuration - LE"](#), on page 88.

## 3.6 Event / Frame configuration - LE

Access:

1. Select "Bluetooth > General > Bluetooth Mode > Bluetooth Low Energy"
2. Select "Channel > Event / Frame Configuration".

The "Event" or "Frame" dialogs vary, depending on the used channel type.

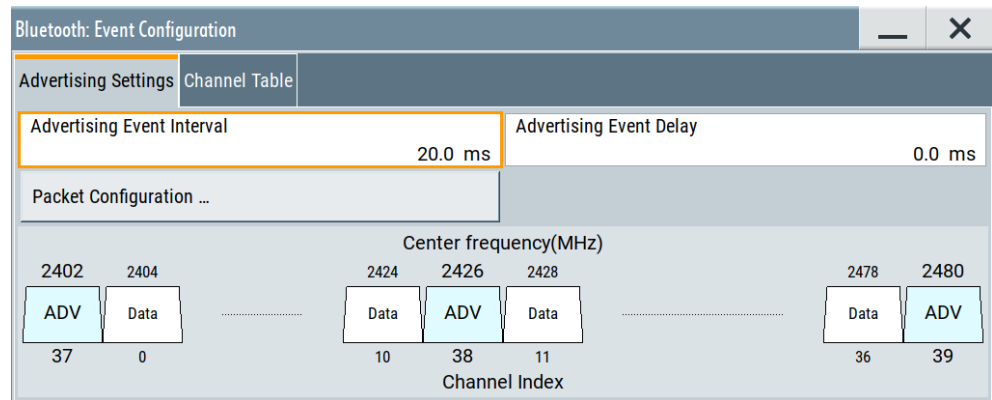


Figure 3-1: Event configuration dialog of the advertising channel type (advertiser)

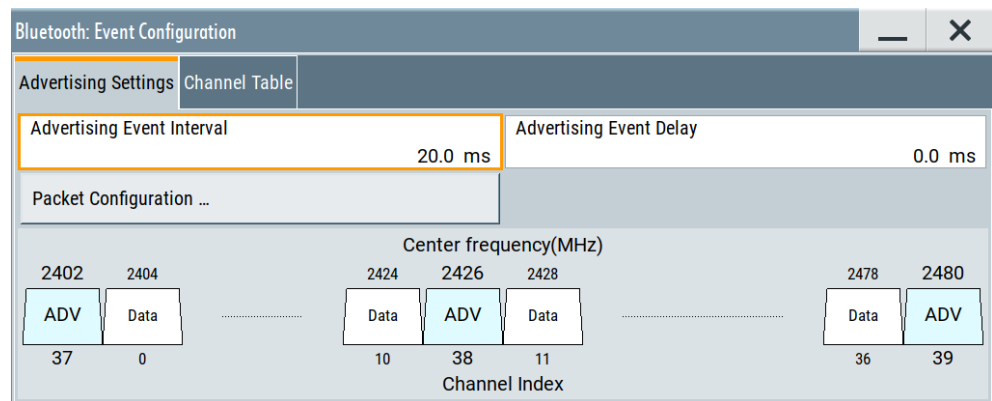


Figure 3-2: Frame configuration dialog of the advertising channel type (scanner)

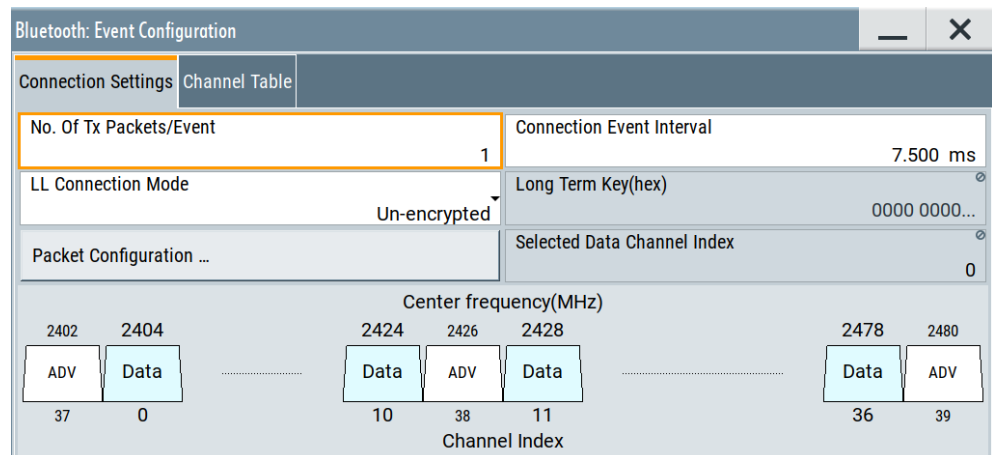


Figure 3-3: Event configuration dialog of the data channel type

The dialogs provide settings to configure event or frame settings. Also, they provide access to the packet configuration dialogs.

The graphics show the distribution of the packets, the physical channel mapping and the channel indices. The channel table gives an overview of the used channels and their assignments.

**Settings**

- [Advertising event / frame settings](#)..... 56
- [Data event settings](#)..... 58
- [Channel table settings](#)..... 61

**3.6.1 Advertising event / frame settings**

The following section describes the parameters necessary for the advertising event or frame configuration.

**Settings:**

<a href="#">Advertising Event Interval</a> .....	56
<a href="#">Periodic Advertising Interval</a> .....	56
<a href="#">Advertising Event Delay</a> .....	56
<a href="#">Scan Window</a> .....	57
<a href="#">Scan Interval</a> .....	57
<a href="#">Advertising Packet Interval</a> .....	57
<a href="#">Transmit Window Offset</a> .....	57
<a href="#">Transmit Window Size</a> .....	58
<a href="#">Packet Configuration</a> .....	58
<a href="#">Channel Table</a> .....	58

**Advertising Event Interval**

Sets the time interval between two consecutive advertising events, regarding the starting points.

**Note:** This parameter is relevant for advertising event configuration and for the packet types ADV\_IND, ADV\_DIRECT\_IND, ADV\_NONCONN\_IND and ADV\_SCAN\_IND. Within the option R&S SMM-K117, the following packet types are also relevant for the setting: ADV\_EXT\_IND, AUX\_ADV\_IND, AUX\_CHAIN\_IND.

Remote command:

For packet type "ADV\_DIRECT\_IND" and duty cycle high:

[\[:SOURCE<hw>\]:BB:BT0oth:ECOnfiguration:ADINterval](#) on page 137

For all others:

[\[:SOURCE<hw>\]:BB:BT0oth:ECOnfiguration:AEINterval](#) on page 137

**Periodic Advertising Interval**

Sets the time interval between the start of two AUX\_SYNC\_IND PDUs from the same advertising set.

Option R&S SMM-K117 is required.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:ECOnfiguration:PCOnfiguration:PAINterval](#) on page 141

**Advertising Event Delay**

Sets a time delay between the start times of two consecutive advertising events. The value is added to the advertising event interval.



**Note:** This parameter is relevant for advertising event configuration and for the packet types ADV\_IND, ADV\_DIRECT\_IND with low duty cycle, ADV\_NONCONN\_IND and ADV\_SCAN\_IND.

Within the option R&S SMM-K117, the following packet types are also relevant for the setting: ADV\_EXT\_IND, AUX\_ADV\_IND, AUX\_CHAIN\_IND.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:AEDelay` on page 137

### Scan Window

Sets the length of the window during which the scanner is operating in the advertising channel.

Note that the scan window is less or equal to the value of the scan interval.

**Note:** This parameter is relevant for advertising frame configuration and for the packet type SCAN\_REQ.

Within the option R&S SMM-K117, the packet type AUX\_SCAN\_REQ is also relevant for the setting.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:SWINdow` on page 143

### Scan Interval

Sets the time interval between the starting points of two consecutive windows during which the scanner is operating in an advertising channel.

**Note:** This parameter is relevant for advertising frame configuration and for the packet type SCAN\_REQ.

Within the option R&S SMM-K117, the packet type AUX\_SCAN\_REQ is also relevant for the setting.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:SIINterval` on page 143

### Advertising Packet Interval

Sets the time interval between packets starting points of two consecutive packets in the advertising channel.

**Note:** This parameter is relevant for advertising frame configuration and for the packet type SCAN\_RSP.

Within the option R&S SMM-K117, the packet type AUX\_SCAN\_RSP is also relevant for the setting.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:APIINterval` on page 138

### Transmit Window Offset

Displays the start point of the transmit window.

**Note:** This parameter is relevant for advertising frame configuration and for the packet type CONNECT\_IND.

Within the option R&S SMM-K117, the following packet types are also relevant for the setting: AUX\_CONNECT\_REQ, AUX\_CONNECT\_RSP.

This parameter is set in the packet configuration dialog, see ["Transmit Window Offset"](#) on page 74.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:WOffset`

on page 180

### Transmit Window Size

Indicates the size of the transmit window, regarding to the start point.

Note that the scan window size is less or equal to the value of the connection interval.

**Note:** This parameter is relevant for advertising frame configuration and for the packet type CONNECT\_IND.

Within the option R&S SMM-K117, the following packet types are also relevant for the setting: AUX\_CONNECT\_REQ, AUX\_CONNECT\_RSP.

The parameter is set in the packet configuration dialog, see ["Transmit Window Size"](#) on page 73.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:WSInfo?` on page 144

### Packet Configuration

Opens the dialog for setting the corresponding packet configuration.

This dialog is described in [Chapter 3.7, "Packet configuration - LE"](#), on page 62.

### Channel Table

Selects the channel to be used for configured packets. The description is covered in [Chapter 3.6.3, "Channel table settings"](#), on page 61.

## 3.6.2 Data event settings

The following section describes the parameters necessary for the data event connection.

### Settings:

<a href="#">No. Of Tx Packets/Event</a> .....	58
<a href="#">Connection Event Interval</a> .....	59
<a href="#">LL Connection Mode</a> .....	59
<a href="#">Long Term key (hex)</a> .....	60
<a href="#">Selected Data Channel Index</a> .....	60

### No. Of Tx Packets/Event

Sets the number of Tx packets per event.

Each connection contains at least one data channel packet. The maximum number of packets per event is determined by the duration of the connection event interval.

**Note:** This parameter is relevant for data event connection settings.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PNUMber` on page 142

### Connection Event Interval

Set the time interval between the start points of two consecutive connection events. Subsequent transmissions within an event are separated by this parameter to separate connecting event starting points in time.

**Note:** This parameter is relevant for data event connection settings and advertising frame configuration with the packet type DATA and all CONTROL\_DATA packet types.

Remote command:

[ :SOURCE<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:CINterval  
on page 163

### LL Connection Mode

Select the link layer connection mode. To provide safe transmission of payload data, the data in the packet can be encrypted. If activated, the payload data follows MIC (message authentication code).

**Note:** This parameter is relevant for data event connection settings.

The following table shows which types of packets can be encrypted and / or unencrypted.

**Table 3-6: Data packet encryption**

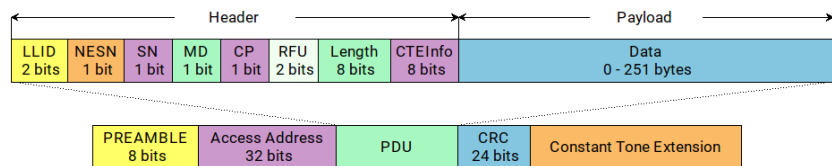
Packet type	encrypted	unencrypted	Packet type	encrypted	unencrypted
DATA	x	x	LL_UNKNOWN_RSP	x	x
LL_CONNECTION_UPDATE_IND	x	x	LL_FEATURE_REQ	x	x
LL_CHANNEL_MAP_IND	x	x	LL_FEATURE_RSP	x	x
LL_TERMINATE_IND	x	x	LL_PAUSE_ENC_REQ	-	x
LL_ENC_REQ	-	x	LL_PAUSE_ENC_RSP	x	-
LL_ENC_RSP	-	x	LL_VERSION_IND	x	x
LL_START_ENC_REQ	-	x	LL_REJECT_IND	x	x
LL_START_ENC_RSP	x	-			

**Table 3-7: Encryption of additional packet types within R&S SMM-K117:**

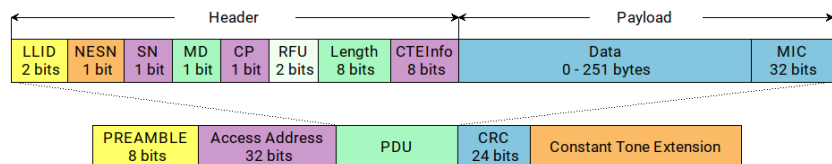
Packet type	encrypted	unencrypted	Packet type	encrypted	unencrypted
LL_PERIPHERAL_FEATURE_REQ	x	x	LL_PHY_RSP	x	x
LL_CONNECTION_PARAM_REQ	x	x	LL_PHY_UPDATE_IND	x	x
LL_CONNECTION_PARAM_RSP	x	x	LL_MIN_USE_CHANNELS_IND	x	x
LL_REJECT_EXT_IND	x	x	LL_CTE_REQ	x	x
LL_PING_REQ	x	x	LL_CTE_RSP	x	x
LL_PING_RSP	x	x	LL_PERIODIC_SYNC_IND	x	x

Packet type	encrypted	unencrypted	Packet type	encrypted	unencrypted
LL_LENGTH_REQ	x	x	LL_CLOCK_ACCURACY_REQ	x	x
LL_LENGTH_RSP	x	x	LL_CLOCK_ACCURACY_RSP	x	x
LL_PHY_REQ	x	x			

"Un-encrypted" Payload data is transmitted without encoding. Example of packet type data:



"Encrypted" The link layer connection runs in encrypted mode. Example of packet type data:



Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:LCMode` on page 138

### Long Term key (hex)

Indicates the time the controller needs to receive the long-term key from the host. After this time, the controller is ready to enter into the last phase of encryption mode setup.

**Note:** This parameter is relevant for data event connection settings. In encrypted mode, the code can be edited.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:LTKey` on page 139

### Selected Data Channel Index

Indicates the number of the first active data channel.

The data channel is selected for each connection event. The Central and Peripheral determine the used data channel by selecting from the list of used channels (see "Channel Table" on page 61).

**Note:** This parameter is relevant for data event connection settings.

Displays the data channel index currently selected.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:SDCI?` on page 142

### 3.6.3 Channel table settings

Access:

1. Follow the directions in [Chapter 3.6, "Event / Frame configuration - LE"](#), on page 54.
2. Select "Channel Table"

	Center Frequency /MHz	Channel Index	Channel Type	Channel State
Channel 0	2 402	37	Advertising	Used
Channel 1	2 426	38	Advertising	Unused
Channel 2	2 480	39	Advertising	Unused

The dialog varies, depending on the used channel type.

The channel table displays all possible channels and with their characteristics and selects the channels to be used for generated packets. Channel frequencies are indicated above the channel table.

#### Settings:

##### Channel Table

The channel table displays configured parameters characterizing the channel and the current state.

Every channel is represented with bit positioned as per the data channel index. LSB represents data channel index 0 and the bit in position 36 represents data channel index 36.

If the channel is used, its channel bit is to be set to '1'. Bit value '0' indicates that the channel is unused.

The bits in positions 37, 38 and 39 must be set to zero upon transmission and ignored upon receipt.

##### "Center Frequency"

Indicates the center frequency of a channel. The graphical representation is displayed above the channel table.

##### "Channel Index"

Indicates the channel index. The graphical representation is displayed above the channel table.

##### "Channel Type"

Indicates the channel type. The graphical representation of possible transmission position is displayed above the channel table.

**"Channel State"**

Specifies the channels to be used for generated packets.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:ACTable:
CHANnel<ch0>:STATe on page 136
```

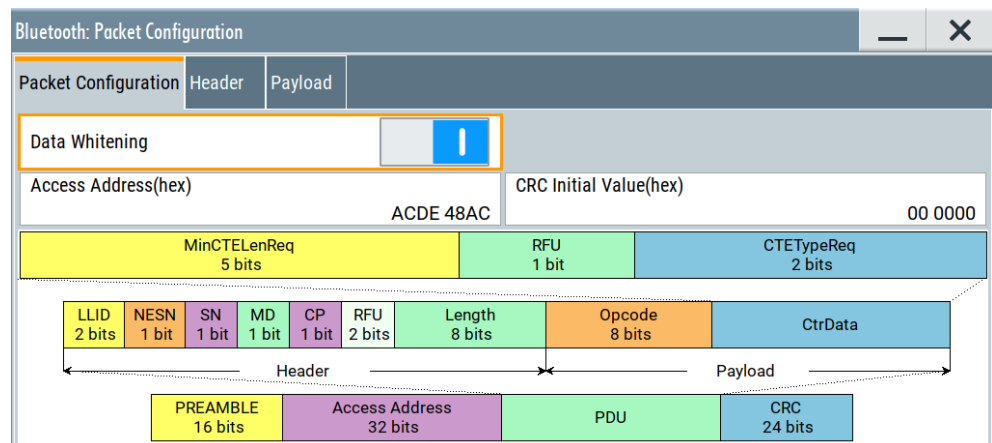
```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:DCTable:
CHANnel<ch0>:STATe on page 136
```

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:
PCOnfiguration:DCMTable:CHANnel<ch0>:STATe
on page 136
```

### 3.7 Packet configuration - LE

Access:

1. Select "Bluetooth" > "General".
2. "Bluetooth Mode" > "Bluetooth LE".
3. Select "Channel" > "Channel Type" > "Advertising"/"Data".
4. Select "Event Configuration"/"Frame Configuration".
5. In the opening dialog, select "Packet Configuration".



The dialog provides settings to configure the parameters of the selected packet type.

**Settings:**

- [General packet configuration](#)..... 63
- [Header configuration](#)..... 63
- [Main payload configuration dialog](#)..... 67
- [Additional payload configuration dialogs](#)..... 83

### 3.7.1 General packet configuration

This section describes the upper part of the "Packet Configuration" tab.

#### Settings:

Data Whitening.....	63
Access Address.....	63

#### Data Whitening

Activates data whitening.

Evenly distributed white noise is ideal for the transmission and real data can be forced to look similar to white noise with different methods called "Data Whitening". Applied to the PDU and CRC fields of all packet types, whitening is used to avoid long equal sequences in the data bit stream.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:DWHitening`  
on page 150

#### Access Address

Sets the access address of the link layer connection.

Bluetooth LE transmissions are based on an interface packet format that consists of a preamble (8 bits) the access address (32 bits), the PDU and CRC (24 bits).

Access address is used to identify communications on a physical channel, and to exclude or ignore packets on different physical channels. The channels are using the same PHY channels in physical proximity.

The structure of access address depends on the packet type:

- Data channel packets  
The access address is a pseudo-random LL connection address, generated by the initiator of the LL connection. The address has to follow some specific rules, which are described in the Bluetooth LE technology.
- Advertising channel packets  
The address is fixed to 01101011011111011001000101110001 with the leftmost bit sent first and being the LSB.

**Note:** This parameter is relevant for all available package types specified in event or frame configuration of a data or advertiser channel.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:AADdress`  
on page 151

### 3.7.2 Header configuration

1. Follow the description in [Chapter 3.7, "Packet configuration - LE"](#), on page 62.

## 2. Select "Header".

Packet Configuration	Header	Payload
NESN Start Value	0	SN Start Value 0
CTEInfo Present	<input type="checkbox"/>	CTEInfo Configuration ...

The tab provides settings to configure the PDU header settings.

**Settings:**

NESN Start Value.....	64
SN Start Value.....	64
Channel Selection.....	64
CRC Initial.....	65
CTEInfo Present.....	65
CTEInfo Configuration.....	65
L CTETime.....	65
L CTEType.....	66
L Antenna Number.....	66
L AntennaX Gain.....	66
Devices Tx/Rx Address Type.....	66

**NESN Start Value**

Sets the start value of the next expected packet from the same device in the LL connection ("Next Expected Sequence Number"). This parameter can be set in the first event. From the second event, this field is not indicated.

**Note:** This parameter is relevant for data event configuration and all data channel packet types except TEST PACKET.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:NSValue`  
on page 152

**SN Start Value**

Sets the sequence number of the packet. This parameter can be set in the first event. From the second event, this field is not indicated.

**Note:** This parameter is relevant for data event configuration and all data channel packet types except TEST PACKET.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:SSValue`  
on page 152

**Channel Selection**

Requires option R&S SMM-K117.

Specifies the algorithm of channel selection signaled via advertising packet types.

"Algorithm #1" Channel selection only for connection events.



"Algorithm #2" Channel selection for connection events and periodic advertising packets.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:CSElection
```

### CRC Initial

Sets the initialization value for the 24 bits cyclic redundancy check (CRC) calculation. A packet has been received correctly, when it has passed the CRC check.

**Note:** This parameter is relevant for data channel types and for advertising packet type CONNECT\_IND.

Within R&S SMM-K117 also for AUX\_CONNECT\_REQ.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:CIValue
```

on page 151

### CTEInfo Present

Activates the CTEInfo field in the header of Bluetooth LE data packets in the LE un-coded PHY.

**Note:** This parameter is relevant for data event configuration and all data channel packet types except TEST PACKET.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:CPresent
```

on page 152

### CTEInfo Configuration

Requires "CTEInfo Present = On".

Accesses the CTEInfo configuration dialog, in which you define CTE length and the CTE method used for direction finding.

CTEInfo Configuration	
CTETime 0.016 ms	CTEType AOD(1us)
Antenna Number 4	
Antenna0 Gain 0.00 dB	Antenna1 Gain 0.00 dB
Antenna2 Gain 0.00 dB	Antenna3 Gain 0.00 dB

CTEInfo		
CTETime 5 bits	RFU 1 bit	CTEType 2 bits

**Note:** This parameter is relevant for data event configuration and all data channel packet types except TEST PACKET.

### CTETime ← CTEInfo Configuration

Sets the CTETime comprising the length of constant tone extension field of the Bluetooth LE PDU.

**Note:** This parameter is relevant for data event configuration and all data channel packet types except TEST PACKET.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:CTIME`  
on page 153

#### **CTEType ← CTEInfo Configuration**

Sets the type of constant tone extension. The type specifies the CTE AoA/AoD method and for AoD the length of the switching and I/Q sampling slots.

**Note:** This parameter is relevant for data event configuration and all data channel packet types except TEST PACKET.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:CTYPE`  
on page 153

#### **Antenna Number ← CTEInfo Configuration**

Requires "CTEType > AoD(1us)/AoD(2us)"

Specifies the number of antennas for angle of departure (AoD) direction finding method. You can select up to four antennas, that are used for direction finding.

**Note:** This parameter is relevant for data event configuration and all data channel packet types except TEST PACKET.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfig:PCOnfig:ANTNumber` on page 154

#### **AntennaX Gain ← CTEInfo Configuration**

Requires "CTEType > AoD(1us)/AoD(2us)"

Specifies the gain of the antenna "AntennaX", where X is 0 to 3 depending on the number of antennas. You can specify the antenna gain information of up for four individual antennas for direction finding.

**Note:** This parameter is relevant for data event configuration and all data channel packet types except TEST PACKET.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfig:PCOnfig:ANTGain<ch0>` on page 154

#### **Devices Tx/Rx Address Type**

Selects the address type of a Bluetooth LE device. Depending on the Bluetooth controller role, either the Tx or Rx or both address types are assigned.

The format of the device address differs depending on the selected address type.

**Note:** This parameter is relevant for advertising event or frame configuration. The Bluetooth controller role and the packet type determine the available entries.

Device address type and corresponding packet types:

- "Tx" for the packet types ADV\_IND, ADV\_DIRECT\_IND, ADV\_NONCONN\_IND, ADV\_SCAN\_IND, SCAN\_REQ, SCAN\_RSP and CONNECT\_IND  
Within R&S SMM-K117 also with the packet types ADV\_EXT\_IND, AUX\_ADV\_IND, AUX\_CHAIN\_IND, AUX\_SYNC\_IND, AUX\_SCAN\_REQ, AUX\_SCAN\_RSP and AUX\_CONNECT\_REQ

- "Rx" for the packet types ADV\_DIRECT\_IND, SCAN\_REQ and CONNECT\_IND

"Public" Allocates a unique 48-bit address to each Bluetooth LE device. Public addresses use an organizationally unique identifier (OUI) obtained from the IEEE registration authority.

"Random" Allocates a 48-bit random static device address to each Bluetooth LE device. A random address is optional. It can be directly generated by the beacon.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:TAType`  
on page 175

### 3.7.3 Main payload configuration dialog

1. Follow the description in [Chapter 3.7, "Packet configuration - LE"](#), on page 62.
2. Select "Payload".

Packet Configuration	Header	Payload	
Transmit Window Size		1.25 ms	Transmit Window Offset 0.00 ms
Connection Event Interval		7.50 ms	Slave Latency 0 Events
LL Connection Timeout		100 ms	Connection Instant 1

The tab provides settings to configure the PDU payload.

#### Settings:

Device Address.....	68
Data Source.....	70
Data Length.....	71
Unknown Type (hex).....	71
Peripheral Latency.....	71
LL Connection Timeout.....	71
Connection Evt Interval.....	71
Connection Instant.....	72
Show / Hide Data Channel (Mapping) Table.....	72
Hop Length.....	72
Random Vector (hex).....	72
Encrypted DIVERSIFIER (hex).....	73
Session Key ID (hex).....	73
Initialization Vector (hex).....	73
Feature Set Length.....	73
Transmit Window Size.....	73
Transmit Window Offset.....	74
Sleep Clock Accuracy.....	74
Error Code.....	74

Version Number(hex).....	75
SubVersion Number(hex).....	75
Company Id(hex).....	75
Advertising Mode.....	75
Target's Device Address.....	75
Extended Header.....	75
L AdvA.....	76
L TargetA.....	76
L CTE Info.....	76
L AdvData Info.....	76
L AuxPtr.....	77
L SyncInfo.....	77
L TxPow.....	77
L ACAD Length.....	77
L ACAD.....	77
L AList / Pattern.....	78
Min. / Max. Interval.....	78
Preferred Periodicity.....	78
Ref. Connection Event Count.....	78
Offset Setting Table.....	79
Max Rx Octets / Max Tx Octets.....	79
Max Rx Time / Max Tx Time.....	79
Rx PHY / Tx PHY.....	79
M_TO_S_PHY/S_TO_M_PHY.....	80
Reject Opcode.....	80
PHYs.....	80
Min Used Channels.....	81
ID(hex).....	81
SyncInfo Configuration.....	81
Connection Event Count.....	81
Last Pa Event Counter.....	81
SID(hex).....	82
Address Type.....	82
PHY.....	82
Sync Connection Event Counter.....	83
MinCTELenReq.....	83
CTETypeReq.....	83
Graph.....	83

### Device Address

Sets the Bluetooth device address. A device address for the LE physical channel is defined in volume 6, part B, section 1.3 of the core specification for Bluetooth wireless technology.

Devices are identified using a device address. Devices use a public device address or a random device address, refer to "[Devices Tx/Rx Address Type](#)" on page 66.

With Bluetooth wireless technology up to the version 4.2, the following address formats are defined:

- "Public Address Type" is the unique 48-bits identity address of each Bluetooth LE device.

The public address is given from the registration authority IEEE and is composed of:

- LSB: 24 bits = company\_assigned
  - MSB: 24 bits = company\_id
  - "Random Address Type" is an optional 48-bits random static device address.
  - "Private Address Type" is a resolvable 48-bits optional address.
- A private address is composed of:
- LSB: 24 bits = hash
  - MSB: 24 bits = random

Since version 5.0, the device address format is in accordance with BD\_ADDR for BR/EDR with the exception that LAP values does not apply. Unless the public device address is also used for a BR/EDR controller.

- **NAP**: Selects non-significant address part. The length of NAP is 16 bits or 4 hexadecimal figures.
- **UAP**: Selects upper address part. The length of UAP is 8 bits or two hexadecimal figures.
- **LAP**: Selects lower address part. The length of LAP is 24 bits or 6 hexadecimal figures.

The NAP+UAP can take any values except the 64 reserved LAP values: #H9E8B00 – #H9E8B3F.

Option R&S SMM-K117 is required for the address formats since Bluetooth version 5.0.

For advertising channel PDU types, refer to "[Payload](#)" on page 24.

Remote command:

Company\_Assigned and Company\_Id in advertisers device address

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:ACID`

on page 158

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:ACASsigned`

on page 158

Company\_Assigned and Company\_Id in scanner's device address

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:SCASsigned`

on page 158

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:SCID`

on page 158

Company\_Assigned and Company\_Id in initiator's device address

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:ICASsigned`

on page 158

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:ICID`

on page 158

NAP+UAP and LAP in advertiser's device address

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:ANUap`

on page 160

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:ALAP`

on page 159

NAP+UAP and LAP in initiators device address

`[ :SOURCE<hw> ] :BB:BT0oth:ECONfiguration:PCONfiguration:INUap`

on page 160

[\[:SOURCE<hw>\]:BB:BT0oth:ECONfiguration:PCONfiguration:ILAP](#)

on page 159

NAP+UAP and LAP in scanners device address

[\[:SOURCE<hw>\]:BB:BT0oth:ECONfiguration:PCONfiguration:SNUap](#)

on page 160

[\[:SOURCE<hw>\]:BB:BT0oth:ECONfiguration:PCONfiguration:SLAP](#)

on page 159

NAP+UAP and LAP in scanner's or initiator's target device address (TargetA) to which the advertisement is directed.

[\[:SOURCE<hw>\]:BB:BT0oth:ECONfiguration:PCONfiguration:TNUap](#)

on page 160

[\[:SOURCE<hw>\]:BB:BT0oth:ECONfiguration:PCONfiguration:TLAP](#)

on page 159

### Data Source

Selects the data source used for the payload.

**Note:** This parameter is relevant for event configuration and packet types DATA, ADV\_IND, ADV\_NONCONN\_IND and ADV\_SCAN\_IND.

The following standard data sources are available:

- "All 0, All 1"  
An internally generated sequence containing 0 data or 1 data.
- "PNxx"  
An internally generated pseudo-random noise sequence.
- "Pattern"  
An internally generated sequence according to a bit pattern.  
Use the "Pattern" box to define the bit pattern.
- "Data List/Select DList"  
A binary data from a data list, internally or externally generated.  
Select "Select DList" to access the standard "Select List" dialog.
  - Select the "Select Data List > navigate to the list file \*.dm\_iqd > Select" to select an existing data list.
  - Use the "New" and "Edit" functions to create internally new data list or to edit an existing one.
  - Use the standard "File Manager" function to transfer external data lists to the instrument.

See also:

- Section "Modulation Data" in the R&S SMM100A user manual.
- Section "File and Data Management" in the R&S SMM100A user manual.
- Section "Data List Editor" in the R&S SMM100A user manual

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:ECONfiguration:PCONfiguration:DATA](#)

on page 163

[\[:SOURCE<hw>\]:BB:BT0oth:ECONfiguration:PCONfiguration:DATA:](#)

[DPATtern](#) on page 164

[\[:SOURCE<hw>\]:BB:BT0oth:ECONfiguration:PCONfiguration:DATA:](#)

[DSElection](#) on page 164

**Data Length**

Enters the payload data length in bytes.

**Note:** This parameter is relevant for event configuration with packet types ADV\_IND, ADV\_NONCONN\_IND and ADV\_SCAN\_IND.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:DLEnGth`  
on page 165

**Unknown Type (hex)**

Enables that an invalid control packet is indicated.

The "CtrType" field indicates the value of the LL control packet that caused the transmission of this packet.

This parameter is relevant for data event configuration with the packet type LL\_UNKNOWN\_RSP (Peripheral).

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:UTYPE`  
on page 179

**Peripheral Latency**

Sets the number of consecutive connection events the Peripheral can ignore for asymmetric link layer connections.

**Note:** This parameter is relevant for data event and advertising frame configuration with the packet types LL\_CONNECTION\_UPDATE\_IND and CONNECT\_IND. Within R&S SMM-K117 also for AUX\_CONNECT\_REQ.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:SLATency`  
on page 177

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:NSLatency`  
on page 173

**LL Connection Timeout**

Defines the maximum time between two correctly received Bluetooth LE packets in the LL connection before the connection is considered lost.

**Note:** This parameter is relevant for data event and advertising frame configuration with the packet types LL\_CONNECTION\_UPDATE\_IND and CONNECT\_IND.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:LCTimeout`  
on page 169

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:NLCTimeout`  
on page 173

**Connection Evt Interval**

Sets new connection event interval between the start points of two consecutive connection events. Subsequent transmissions within an event are separated by this parameter to separate connecting event starting points in time.

**Note:** This parameter is relevant for data event connection settings and advertising frame configuration with the packet types LL\_CONNECTION\_UPDATE\_IND and CONNECT\_IND.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:NCINterval`  
on page 172

### Connection Instant

Sets a connection instant for indicating the connection event at which the new connection parameters are taken in use.

Both the Central and the Peripheral have a 32-bit connection event counter per LL connection. It is reset to zero on the first connection event of the LL connection and incremented by one on every elapsed connection event interval of the LL connection.

**Note:** This parameter is relevant for data event configuration with the packet types LL\_CONNECTION\_UPDATE\_IND and LL\_CHANNEL\_MAP\_IND.

Within the option R&S SMM-K117, the following data packet types are also relevant for the setting: LL\_PHY\_UPDATE\_IND.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:CINstant`  
on page 163

### Show / Hide Data Channel (Mapping) Table

In data event and advertising frame configuration with the packet types LL\_CHANNEL\_MAP\_IND and CONNECT\_IND, calls / hides the channel map table that displays the used channels and their parameters.

The channel table is described in [Chapter 3.6.3, "Channel table settings"](#), on page 61.

Remote command:

n.a.

### Hop Length

Sets the difference from the current channel to the next channel.

The Central and the Peripherals determine the data channel in use for every connection event from the channel map. Hop\_length is set for the LL connection and communicated in the CONNECT\_IND packets.

**Note:** This parameter is relevant for data event and advertising frame configuration with the packet type CONNECT\_IND.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:HLENgth`  
on page 169

### Random Vector (hex)

Sets the random vector of the Central for device identification.

The parameter is an initialization vector provided by the host in the HCI\_ULP\_Start\_Encryption command.

**Note:** This parameter is relevant for data event configuration with the packet type LL\_ENC\_REQ.



Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:RVEctor`  
on page 176

#### **Encrypted DIVERSifier (hex)**

Sets the encrypted diversifier of the Central for device identification. The parameter is an initialization vector provided by the host in the HCI\_ULP\_Start\_Encryption command.

**Note:** This parameter is relevant for data event configuration with the packet type LL\_ENC\_REQ.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:EDIVersifier`  
on page 166

#### **Session Key ID (hex)**

Sets the portion of the Central or the portion of the Peripheral of the session key diversifier (SKDm/SKDs).

**Note:** This parameter is relevant for data event configuration with the packet types LL\_ENC\_REQ (Central) and LL\_ENC\_RSP (Peripheral).

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:MSKD`  
on page 170

#### **Initialization Vector (hex)**

Sets the portion of the Central or the portion of the Peripheral of the initialization vector(IVm/IVs).

**Note:** This parameter is relevant for data event configuration with the packet types LL\_ENC\_REQ (Central) and LL\_ENC\_RSP (Peripheral).

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:MIVector`  
on page 170

#### **Feature Set Length**

Specifies the length of feature set for Central (LL\_FEATURE\_REQ) or Peripheral.

For feature set setting within the option R&S SMM-K117, refer to "[FeatureSet Configuration](#)" on page 87.

**Note:** This parameter is signaled via LL\_FEATURE\_REQ (Central) and LL\_FEATURE\_RSP (Peripheral).

Within the option R&S SMM-K117, it is signaled also via LL\_PERIPHERAL\_FEATURE\_REQ.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:FSLength`  
on page 169

#### **Transmit Window Size**

Sets the size of the transmit window, regarding to the start point.

Note that the scan window size is less or equal to the value of the connection interval, see "[Connection Evt Interval](#)" on page 71.

**Note:** This parameter is relevant for advertising frame configuration for the packet type CONNECT\_IND and for data packet type LL\_CONNECTION\_UPDATE\_IND. Within the option R&S SMM-K117, the following data packet types are also relevant for the setting: AUX\_CONNECT\_REQ, AUX\_CONNECT\_RSP.

This parameter is also indicated in the "Frame Configuration Dialog".

Remote command:

For advertising channels:

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:WSize
```

on page 180

For data channels:

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:NWSize
```

on page 174

### Transmit Window Offset

Sets the start point of the transmit window.

**Note:** This parameter is relevant for advertising frame configuration and for the packet types CONNECT\_IND and LL\_CONNECTION\_UPDATE\_IND.

This parameter is also indicated in the "Frame Configuration Dialog".

Remote command:

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:WOffset
```

on page 180

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:NWOffset
```

on page 173

### Sleep Clock Accuracy

Defines the clock accuracy of the Central with specified encoding. This parameter is used by the Peripheral to determine required listening windows in the LL connection. It is a controller design parameter known by the controller.

**Note:** This parameter is relevant for advertising frame configuration and the packet types CONNECT\_IND and LL\_PERIODIC\_SYNC\_IND.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:SCACcuracy
```

on page 177

### Error Code

Sets the error code value to inform the remote device why the connection is about to be terminated for a LL\_TERMINATE\_IND packet.

For LL\_REJECT\_IND packet, this parameter is used when a request was rejected. An 8-bit value is set.

**Note:** This parameter is relevant for data frame configuration and the packet type LL\_TERMINATE\_IND and LL\_REJECT\_IND.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:ECODE
```

on page 165

**Version Number(hex)**

Sets the version of the Bluetooth controller specification (8 bits).

**Note:** This parameter is relevant for data frame configuration and the packet type LL\_VERSION\_IND.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:VNumber`  
on page 179

**SubVersion Number(hex)**

Sets a unique 16-bit value for each implementation or revision of an implementation of the Bluetooth controller.

**Note:** This parameter is relevant for data frame configuration and for the packet type LL\_VERSION\_IND.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:SVNumber`  
on page 178

**Company Id(hex)**

Sets the 16-bit company identifier of the manufacturer of the Bluetooth controller.

**Note:** This parameter is relevant for data frame configuration and for the packet type LL\_VERSION\_IND.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:CID`  
on page 162

**Advertising Mode**

Requires option R&S SMM-K117.

Indicates the mode of the advertisement. All modes defined in specification are supported.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:AMode`  
on page 160

**Target's Device Address**

TargetA parameter, refer to "[Device Address](#)" on page 68.

**Extended Header**

Requires option R&S SMM-K117.

Enables the extended header for advertising packets with scanning PDUs.

If enabled, the following parameters are displayed in the table below. These parameters are signaled via ADV\_EXT\_IND, AUX\_ADV\_IND, AUX\_SCAN\_RSP, AUX\_SYNC\_IND, AUX\_CHAIN\_IND, AUX\_CONNECT\_RSP.

Extended Header											
AdvA (NAP,UAP)	(LAP)	TargetA (NAP,UAP)	(LAP)	CTE Info	AdvData Info	Aux Ptr	Sync Info	TxPow (dBm)	ACAD Length	ACAD	AList / Pattern
<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
AC DE48	00 0080	AC DE48	00 0080	Conf...	Conf...	Conf...	Conf...	0	26	PN 9	

Remote command:

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:EHHeader:
STATE on page 166
```

#### AdvA ← Extended Header

If enabled, the R&S SMM100A includes the signaling of non-significant advertising address part (NAP), upper address part (UAP) and lower address part (LAP). The settings of NAP, UAP and LAP are covered in the section "[Device Address](#)" on page 68.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:EHFlags:
ADDRESS:STATE on page 166
```

#### TargetA ← Extended Header

Enables the signaling of non-significant address part (NAP), upper address part (UAP) and lower address part (LAP). The settings of NAP, UAP and LAP are covered in the section "[Device Address](#)" on page 68.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:EHFlags:
TADDRESS:STATE on page 168
```

#### CTE Info ← Extended Header

Activates the CTEInfo field in the header of Bluetooth LE data packets in the LE uncodded PHY.

**Note:** This parameter is relevant for data event configuration and all data channel packet types except TEST PACKET.

"Config" accesses the CTEInfo configuration dialog to define the CTE length and the CTE method for direction finding. See "[CTEInfo Configuration](#)" on page 65.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:EHFlags:
CINFO:STATE on page 167
```

#### AdvData Info ← Extended Header

Enables the signaling of advertising data information consisting of "Advertising Data ID" and "Advertising Set ID". The setting is covered in the section "[AdvDataInfo Configuration](#)" on page 84.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:EHFlags:
ADINFO:STATE on page 167
```

**AuxPtr ← Extended Header**

Enables the secondary advertising channel. The setting is covered in the section "[AuxPtr Configuration](#)" on page 84.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:EHFLags:
APTR:STATe on page 167
```

**SyncInfo ← Extended Header**

Enables the signaling of SyncInfo. The presence of the SyncInfo field indicates the presence of a periodic advertisement. The setting is covered in the section "[SyncInfo Configuration](#)" on page 85.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:EHFLags:
SINFo:STATe on page 167
```

**TxPow ← Extended Header**

Enables and sets the signaling of required transmit power.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:EHFLags:
TPOWer:STATe on page 168
```

```
[ :SOURCE<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:TPOWer
on page 178
```

**ACAD Length ← Extended Header**

Specifies the length of additional controller advertising data (ACAD) field.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:ALENght
on page 159
```

**ACAD ← Extended Header**

Specifies the pattern used for additional controller advertising data (ACAD).

The following standard data sources are available:

- "All 0, All 1"  
An internally generated sequence containing 0 data or 1 data.
- "PNxx"  
An internally generated pseudo-random noise sequence.
- "Pattern"  
An internally generated sequence according to a bit pattern.  
Use the "Pattern" box to define the bit pattern.
- "Data List/Select DList"  
A binary data from a data list, internally or externally generated.  
Select "Select DList" to access the standard "Select List" dialog.
  - Select the "Select Data List > navigate to the list file \*.dm\_iqd > Select" to select an existing data list.
  - Use the "New" and "Edit" functions to create internally new data list or to edit an existing one.
  - Use the standard "File Manager" function to transfer external data lists to the instrument.

See also:

- Section "Modulation Data" in the R&S SMM100A user manual.
- Section "File and Data Management" in the R&S SMM100A user manual.
- Section "Data List Editor" in the R&S SMM100A user manual

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:ACAD`  
on page 156

#### **AList / Pattern ← Extended Header**

Specifies the path of internal data list for **ACAD** = "Data List" or

Sets the user-defined ACAD pattern for **ACAD** = "Pattern".

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:ACAD:`  
`APATtern` on page 157

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:ACAD:`  
`ASElection` on page 157

#### **Min. / Max. Interval**

Specifies the minimum / maximum allowed connection interval.

**Note:** These parameters are signaled via LL\_CONNECTION\_PARAM\_REQ and LL\_CONNECTION\_PARAM\_RSP.

Option R&S SMM-K117 is required.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:MNInterval`  
on page 139

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:MXInterval`  
on page 140

#### **Preferred Periodicity**

Requires option R&S SMM-K117.

Specifies the preferred periodicity. This value is typically a multiple of the value of the connection interval.

**Note:** This parameter is signaled via LL\_CONNECTION\_PARAM\_REQ and LL\_CONNECTION\_PARAM\_RSP.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:`  
`PPERiodicity` on page 141

#### **Ref. Connection Event Count**

Requires option R&S SMM-K117.

Specifies connection event counter relative to the calculation of all valid offset fields Offset0 to Offset5.

**Note:** This parameter is signaled via LL\_CONNECTION\_PARAM\_REQ and LL\_CONNECTION\_PARAM\_RSP.

See also [Offset Setting Table](#).

Remote command:

[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:RCECount  
on page 142

### Offset Setting Table

Requires option R&S SMM-K117.

Specifies the possible positions of the anchor points of the LE connection with the updated connection parameters relative to the [Ref. Connection Event Count](#).

**Note:** These parameters are signaled via LL\_CONNECTION\_PARAM\_REQ and LL\_CONNECTION\_PARAM\_RSP.

Remote command:

[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:  
OFFSet<ch0>:STATe on page 140  
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:  
OFFSet<ch0>:VALue on page 140

### Max Rx Octets / Max Tx Octets

Requires option R&S SMM-K117.

Specifies the maximum allowed payload length of a packet to be received (Rx) or transmitted (Tx).

**Note:** These parameters are signaled via LL\_LENGTH\_REQ and LL\_LENGTH\_RSP.

Remote command:

[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:MROctets  
on page 170  
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:MTOctets  
on page 170

### Max Rx Time / Max Tx Time

Requires option R&S SMM-K117.

Specifies the maximum allowed time to receive (Rx) or transmit (Tx) a packet.

**Note:** These parameters are signaled via LL\_LENGTH\_REQ and LL\_LENGTH\_RSP.

Remote command:

[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:MRTime  
on page 171  
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:MTTime  
on page 171

### Rx PHY / Tx PHY

Requires option R&S SMM-K117.

Specifies preferred physical layers in receive (Rx) and transmit (Tx) direction. For permitted PHYs, refer to [Table 2-22](#).

**Note:** These parameters are signaled via LL\_PHY\_REQ and LL\_PHY\_RSP.

Remote command:

[ :SOURce<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:RPHYs:L1M:STATE on page 176

[ :SOURce<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:RPHYs:L2M:STATE on page 176

[ :SOURce<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:RPHYs:LCOD:STATE on page 176

[ :SOURce<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:TPHYs:L1M:STATE on page 176

[ :SOURce<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:TPHYs:L2M:STATE on page 176

[ :SOURce<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:TPHYs:LCOD:STATE on page 176

### M\_TO\_S\_PHY/S\_TO\_M\_PHY

Requires option R&S SMM-K117.

Specifies the physical layers to be used in Central-to-Peripheral (C\_TO\_P) and Peripheral-to-Central (P\_TO\_C) direction. For permitted PHYs, refer to [Table 2-22](#).

For Bluetooth core specifications version 5.2 and lower, the notations "M\_TO\_S\_PHY"/"S\_TO\_M\_PHY" are used.

**Note:** This parameter is signaled via LL\_PHY\_UPDATE\_IND.

Remote command:

[ :SOURce<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:MTSPhy:L1M:STATE on page 172

[ :SOURce<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:MTSPhy:L2M:STATE on page 172

[ :SOURce<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:MTSPhy:LCOD:STATE on page 172

[ :SOURce<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:STMPHy:L1M:STATE on page 172

[ :SOURce<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:STMPHy:L2M:STATE on page 172

[ :SOURce<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:STMPHy:LCOD:STATE on page 172

### Reject Opcode

Requires option R&S SMM-K117.

Specifies the Opcode of rejected LL control PDU. For Opcode, refer to [Table 2-20](#).

**Note:** This parameter is signaled via LL\_REJECT\_EXT\_IND.

Remote command:

[ :SOURce<hw> ] :BB:BT00th:ECOnfiguration:PCOnfiguration:ROPCode on page 175

### PHYs

Requires option R&S SMM-K117.



Specifies the physical layers for which the Peripheral has a [Min Used Channels](#) requirement.

**Note:** These parameters are signaled via LL\_MIN\_USED\_CHANNELS\_IND.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:PHYS:L1M:STATe` on page 174

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:PHYS:L2M:STATe` on page 174

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:PHYS:LCOD:STATe` on page 174

### Min Used Channels

Requires option R&S SMM-K117.

Specifies the minimum number of channels to be used on the specified [PHYs](#).

**Note:** This parameter is signaled via LL\_MIN\_USED\_CHANNELS\_IND.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:MUChannels` on page 171

### ID(hex)

Specifies the ID of the identifier specified by the Host in the CtrData field. The value is set in hexadecimal representation.

**Note:** This parameter is signaled via LL\_PERIODIC\_SYNC\_IND control data PDU.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:ID` on page 181

### SyncInfo Configuration

Accesses the "SyncInfo Configuration" dialog. See "[SyncInfo Configuration](#)" on page 85.

### Connection Event Count

Specifies the connEventCount field in the CtrData field.

The count value is specified within the following range:

$\text{currEvent} - 2^{14} < \text{connEventCount} < \text{currEvent} + 2^{14}$

CurrEvent is the counter value for the connection event during (re-) transmission of the LL\_PERIODIC\_SYNC\_IND PDU.

**Note:** This parameter is signaled via LL\_PERIODIC\_SYNC\_IND control data PDU.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:CECount` on page 181

### Last Pa Event Counter

Requires option R&S SMM-K117.

Specifies the lastPaEventCounter field in the CtrData field.

The lastPaEventCounter value is typically set to the PaEventCounter value in the AUX\_SYNC\_IND PDU.

Specified are the following values for lastPaEventCounter and EventCounter:

- Equal values
- Values with a difference of 1 (modulo 65536)
- Values representing LL\_PERIODIC\_SYNC\_IND and AUX\_SYNC\_IND timing of less than 5 seconds

**Note:** This parameter is signaled via LL\_PERIODIC\_SYNC\_IND control data PDU.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:LPECOUNTER`  
on page 181

### SID(hex)

Requires option R&S SMM-K117.

Specifies the SID field in the CtrData field. The value is set in hexadecimal representation.

The SID is typically set to the Advertising SID subfield of the advertising set pointing to periodic advertising.

**Note:** This parameter is signaled via LL\_PERIODIC\_SYNC\_IND control data PDU.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:SID`  
on page 183

### Address Type

Requires option R&S SMM-K117.

Specifies the address type in the CtrData field.

**Note:** This parameter is signaled via LL\_PERIODIC\_SYNC\_IND control data PDU.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:ATYPE`  
on page 180

### PHY

Requires option R&S SMM-K117.

Specifies the PHY field in the CtrData field. The value is set in hexadecimal representation.

The PHY information is used to indicate the PHY type used by periodic advertising. The selection is exclusive, i.e. enabling one PHY disables the other enabled PHY.

**Note:** This parameter is signaled via LL\_PERIODIC\_SYNC\_IND control data PDU.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:PHY:L1M:STATE` on page 182

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:PHY:L2M:STATE` on page 182

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:PHY:LCOD:STATE` on page 182

**Sync Connection Event Counter**

Requires option R&S SMM-K117.

Sets the event counter for the sync connection.

**Note:** This parameter is signaled via LL\_PERIODIC\_SYNC\_IND control data PDU.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:SCECounter`  
on page 183

**MinCTELenReq**

Requires option R&S SMM-K117.

Specifies the minimum CTE length in the CtrData field.

**Note:** This parameter is signaled via LL\_CTE\_REQ control data PDU.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:MCLReq`  
on page 181

**CTETypeReq**

Requires option R&S SMM-K117.

Specifies the minimum CTE length in the CtrData field.

**Note:** This parameter is signaled via LL\_CTE\_REQ control data PDU.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:CTReq`  
on page 153  
`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:MCLReq`  
on page 181

**Graph**

The figure in the packet configuration dialog shows the packet structure of the currently selected packet type.

**3.7.4 Additional payload configuration dialogs**

Option R&S SMM-K117.

The following additional dialogs can be accessed from the payload tab of the packet configuration dialog.

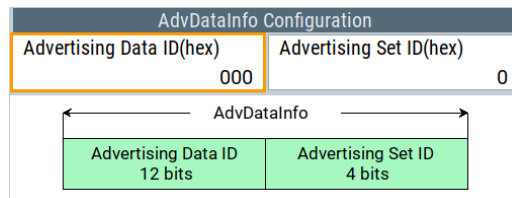
<a href="#">AdvDataInfo Configuration</a> .....	84
<a href="#">AuxPtr Configuration</a> .....	84
L <a href="#">Channel Table</a> .....	84
L <a href="#">Clock Accuracy</a> .....	85
L <a href="#">Offset Units</a> .....	85
L <a href="#">Aux Offset</a> .....	85
L <a href="#">AUX PHY</a> .....	85
<a href="#">SyncInfo Configuration</a> .....	85
L <a href="#">Sync Packet Offset</a> .....	86
L <a href="#">Offset Units</a> .....	86

- L Offset Adjust..... 86
- L Periodic Adv Interval..... 86
- L Secondary Advertising Channel Map Table..... 86
- L Sleep Clock Accuracy..... 86
- L Access Address..... 86
- L CRC Initial Value..... 86
- L Event Counter..... 86
- FeatureSet Configuration..... 87

**AdvDataInfo Configuration**

Specifies advertising data information consisting of "Advertising Data ID" and "Advertising Set ID". The structure of data is displayed also graphically.

These parameters are signaled within an extended header, refer to "Extended Header" on page 75.



Remote command:

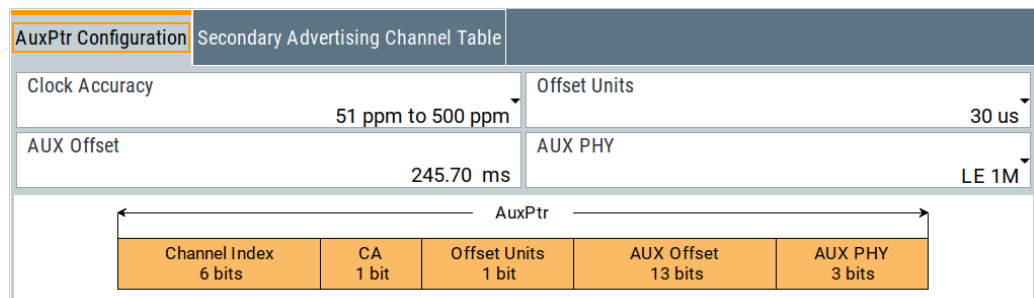
`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:ADID`  
on page 158

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:ASID`  
on page 162

**AuxPtr Configuration**

The presence of the AuxPtr field indicates that some or all advertisement data is in a subsequent auxiliary packet. The contents of the AuxPtr field describe this packet. The structure of data is displayed also graphically.

These parameters are signaled within an extended header, refer to "Extended Header" on page 75.



**Channel Table ← AuxPtr Configuration**

Selects the channel to be used as secondary advertising channel (auxiliary packet). Every channel is represented with a bit positioned as per the data channel index. The settings are identical with data channel table described in Chapter 3.6.3, "Channel table settings", on page 61.

**Clock Accuracy ← AuxPtr Configuration**

Specifies the clock accuracy of the advertiser used between the packet containing this data and the auxiliary packet.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:CACCuracy`  
on page 162

**Offset Units ← AuxPtr Configuration**

Indicates the units used by the [Aux Offset](#) parameter.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:AOUNits`  
on page 161

**Aux Offset ← AuxPtr Configuration**

Specifies the time from the start of the packet containing the AuxPtr field to the approximate start of the auxiliary packet.

The parameter unit of time is specified by the [Offset Units](#). The offset is determined by multiplying the value by the unit. Set the value at least to the length of the packet plus 300 µs.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:AOFFset`  
on page 161

**AUX PHY ← AuxPtr Configuration**

Specifies the physical layer used to transmit the auxiliary packet.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:APHY`  
on page 161

**SyncInfo Configuration**

The presence of the SyncInfo field indicates the presence of a periodic advertisement (using AUX\_SYNC\_IND PDUs). The contents of the SyncInfo field describe this periodic advertisement. The structure of data is displayed also graphically.

The parameters are configurable via AUX\_ADV\_IND. They are signaled within an extended header, refer to "[Extended Header](#)" on page 75.

SyncInfo Configuration		Secondary Advertising Channel Map Table							
Sync Packet Offset	245.70 ms	Offset Units	30 us						
Offset Adjust	<input type="checkbox"/>	Periodic Adv Interval	20.00 ms						
Sleep Clock Accuracy	251 ppm - 500 ppm	Access Address(hex)	ACDE 48AC						
CRC Initial Value(hex)	00 0000	Event Counter	0 Events						
← SyncInfo →									
Sync Packet Offset 13 bits	Offset Units 1 bit	Offset Adjust 1 bit	RFU 1 bit	Interval 16 bits	ChM 37 bits	SCA 3 bits	AA 32 bits	CRCInit 24 bits	Event Counter 16 bits

**Sync Packet Offset ← SyncInfo Configuration**

Specifies the time from the start of the AUX\_ADV\_IND packet containing the SyncInfo field to the start of the AUX\_SYNC\_IND packet.

The sync packet offset consists of multiples of the set [Offset Units](#).

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:SPOffset`  
on page 178

**Offset Units ← SyncInfo Configuration**

Sets the offset units of the [Sync Packet Offset](#) parameter.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:SOUNits`  
on page 177

**Offset Adjust ← SyncInfo Configuration**

Adjusts the "Sync Packet Offset" automatically to the next value, which is a multiple of the "Offset Units".

If "Offset Adjust = On", the "Sync Packet Offset" is set to 2.4567 s and "Offset Units = 300 μs".

If "Offset Units > 30 μs", the offset adjust is deactivated.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:OADJust`  
on page 174

**Periodic Adv Interval ← SyncInfo Configuration**

Refer to "[Periodic Advertising Interval](#)" on page 56.

**Secondary Advertising Channel Map Table ← SyncInfo Configuration**

Selects the channel to be used as secondary advertising channel (auxiliary packet). Every channel is represented with a bit positioned as per the data channel index. The settings are identical with data channel table described in [Chapter 3.6.3, "Channel table settings"](#), on page 61.

**Sleep Clock Accuracy ← SyncInfo Configuration**

Refer to "[Sleep Clock Accuracy](#)" on page 74.

**Access Address ← SyncInfo Configuration**

Refer to "[Access Address](#)" on page 63.

**CRC Initial Value ← SyncInfo Configuration**

Refer to "[CRC Initial](#)" on page 65.

**Event Counter ← SyncInfo Configuration**

Counts the AUX\_SYNC\_IND packets that the SyncInfo field describes.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:ECOUNTER`  
on page 165

### FeatureSet Configuration

Bit position	Link Layer Feature	Valid
0	LE Encryption	<input checked="" type="checkbox"/>
1	Connection Parameters Request Procedure	<input checked="" type="checkbox"/>
2	Extended Reject Indication	<input checked="" type="checkbox"/>

Specifies the supported feature set. Information is transmitted via LL\_FEATURE\_REQ, LL\_FEATURE\_RSP, and LL\_PERIPHERAL\_FEATURE\_REQ.

Configurable are the features as listed in [Table 3-8](#).

**Table 3-8: Link layer features: Bit number and feature**

Bit	Link layer feature	Bit	Link layer feature
0	LE encryption	14	Channel selection algorithm #2
1	Connection parameter request procedure	15	LE power class 1
2	Extended reject indication	16	Minimum Number of Used Channels procedure
3	Peripheral-initiated features exchange	17	Connection CTE request
4	LE ping	18	Connectionless CTE response
5	LE data packet length extension	19	Connectionless CTE transmitter
6	LL privacy	20	Connectionless CTE receiver
7	Extended scanner filter policies	21	Antenna switching during CTE transmission (AoD)
8	LE 2M PHY	22	Antenna switching during CTE reception (AoD)
9	Stable modulation index - transmitter	23	Receiving constant tone extensions
10	Stable modulation index - receiver	24	Periodic advertising Sync transfer - sender
11	LE coded PHY	25	Periodic advertising Sync transfer - recipient
12	LE extended advertising	26	Sleep clock accuracy updates
13	LE periodic advertising	27	Remote public key validation

Remote command:

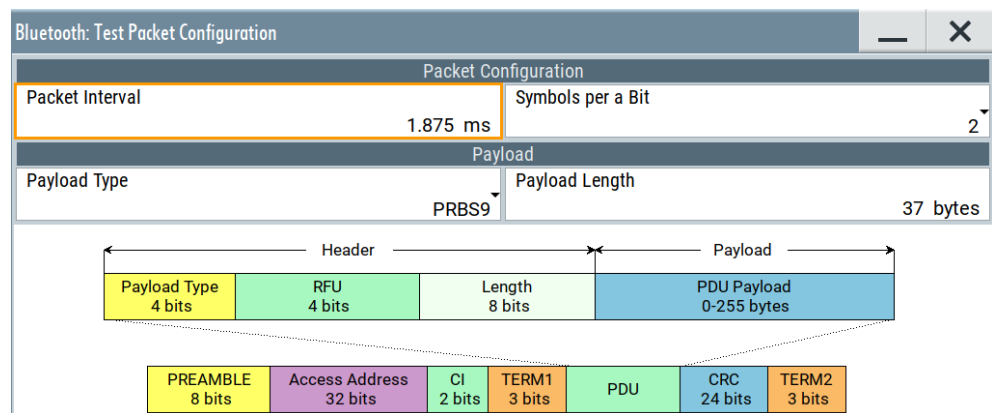
```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:
FSBit<ch0>:STATe on page 168
```

## 3.8 Test packet configuration - LE

Access:

1. Select "Bluetooth" > "General".
2. "Bluetooth Mode" > "Bluetooth Low Energy".
3. Select "Channel" > "Packet Type" > "TEST PACKET".
4. Select "Test Packet Configuration"

The dialog provides settings to configure test packets. The graphic shows the test packet structure and fields.



Settings:

Packet Configuration.....	88
Packet Interval.....	88
Symbols per a Bit.....	89
Header.....	89
Payload.....	89
Payload Type.....	89
Payload Length.....	89

### Packet Configuration

In this section, configure general packet settings.

#### Packet Interval

Sets the time interval between two consecutive test packets, regarding the starting points.

Test packet interval

**Note:** This parameter is relevant for test packet types only.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:DTTest:TPConfiguration:TPINterval`

on page 184



**Symbols per a Bit**

Requires option R&S SMM-K117 and "Packet Format" > "LE Coded".

Specifies a coding for LE coded packets. The specification for Bluetooth wireless technology defines two values S for forward error correction: S = 2 symbols/bit and S = 8 symbols/bit.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:SPBit  
on page 184
```

**Header**

In this section, configure CTE header settings of the test packet. See "[CTEInfo Present](#)" on page 65 and "[CTEInfo Configuration](#)" on page 65.

**Payload**

In this section, configure payload type and payload length.

**Payload Type**

Selects the data source used for the payload test packets.

**Note:** This parameter is relevant for test packet types only.

"PRBS 9, 15" Pseudo-random bit sequences of the length 9 or 15 - transmission of identical packet series.

"Predefined pattern"

11110000, 10101010, 11111111, 00000000, 00001111, or 01010101

Remote command:

```
[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TPConfiguration:UPSource  
on page 185
```

**Payload Length**

Sets the test packet payload length.

Remote command:

```
[ :SOURCE<hw> ] :BB:BT0oth:DTTest:TPConfiguration:UPLength  
on page 184
```

## 4 Signal control and signal characteristics

This section lists settings provided for configuring the baseband filter and configuring power ramping of bluetooth bursts. Also settings are listed for defining the signal generation start and for generating signals necessary for synchronization with other instruments.

It covers the following topics:

- [Filter/clipping settings](#).....90
- [Power ramping settings](#).....96
- [Trigger settings](#)..... 97
- [Marker settings](#).....102
- [Clock settings](#)..... 104
- [Local and global connectors settings](#)..... 105

### 4.1 Filter/clipping settings

Access:

- ▶ Select "Baseband > Bluetooth > General > Filter/Clipping".

The dialog comprises the settings, necessary to configure the baseband filter, the modulation settings and to enable clipping.

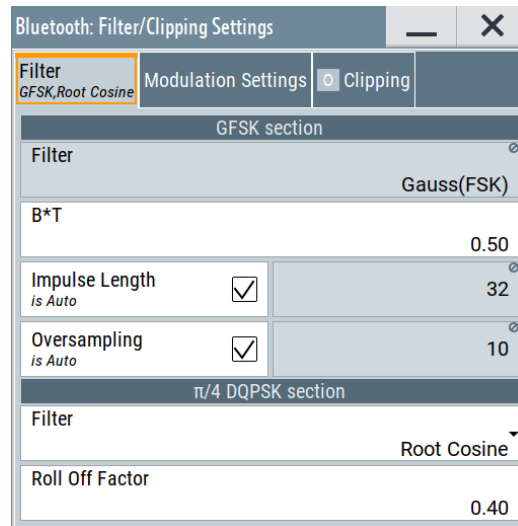
**Settings:**

- 4.1.1 Filter settings.....91
- 4.1.2 Modulation settings..... 93
- 4.1.3 Clipping settings.....94

## 4.1.1 Filter settings

Access:

- ▶ Select "Baseband > Bluetooth > General > Filter/Clipping > Filter".



The dialog provides settings to configure the baseband filter.

### Settings:

GFSK section.....	91
L Filter.....	91
L Roll Off Factor / BxT.....	91
L Impulse Length.....	92
L Oversampling.....	92
π/4 DQPSK / 8DPSK section.....	92
L Filter.....	92
L Roll Off Factor / BxT.....	92
L Cut Off Frequency Factor.....	93

### GFSK section

Provides filter settings for the GFSK section.

#### Filter ← GFSK section

Displays the filter used for GFSK section.

For the  $\pi/4$  DQPSK section or 8 DPSK section you can set filters, see "Filter" on page 92.

Remote command:

n.a.

#### Roll Off Factor / BxT ← GFSK section

Sets the filter parameter.

The filter parameter ("Roll off Factor" or "BxT") depends on the currently selected filter type. This parameter is preset to the default for each of the predefined filters.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:APCO25` on page 189

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:COSine` on page 189

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:FGAuss` on page 189

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:GAUSSs` on page 190

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:LPASSs` on page 190

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:PGAuss` on page 190

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:RCOSine` on page 191

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:SPHase` on page 191

### Impulse Length ← GFSK section

Displays the number of filter tabs.

If enabled, the most sensible parameter values are selected. The value depends on the coherence check.

Disable it to set the values manually.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:ILENgtH:AUTO [ :STATe ]` on page 187

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:ILENgtH` on page 187

### Oversampling ← GFSK section

Sets the upsampling factor.

If enabled, the most sensible parameter values are selected. The value depends on the coherence check.

Disable it to change the value manually.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:OSAMpling:AUTO [ :STATe ]` on page 188

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:OSAMpling` on page 188

### $\pi/4$ DQPSK / 8DPSK section

Provides filter settings for the  $\pi/4$  DQPSK / 8DPSK section.

#### Filter ← $\pi/4$ DQPSK / 8DPSK section

Selects the filter used for DQPSK/8DPSK sections with EDR packets.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:TYPE` on page 187

#### Roll Off Factor / BxT ← $\pi/4$ DQPSK / 8DPSK section

Sets the filter parameter.

The filter parameter ("Roll off Factor" or "BxT") depends on the currently selected filter type. This parameter is preset to the default for each of the predefined filters.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:APCO25` on page 189

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:COSine` on page 189

`[ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:FGAuss` on page 189

[ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:GAUSS on page 190  
 [ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:LPASs on page 190  
 [ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:PGAuss on page 190  
 [ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:RCOSine on page 191  
 [ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:SPHase on page 191

#### Cut Off Frequency Factor ← $\pi/4$ DQPSK / 8DPSK section

Requires a lowpass filter.

Sets the value for the cutoff frequency factor. The cutoff frequency of the filter can be adjusted to reach spectrum mask requirements.

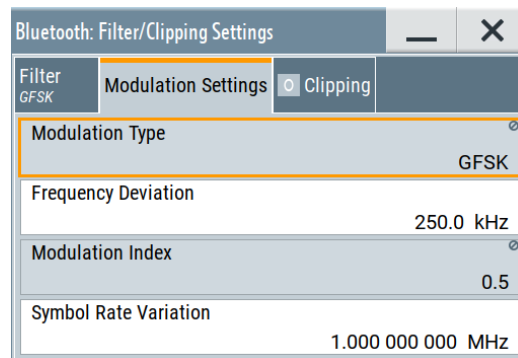
Remote command:

[ :SOURce<hw> ] :BB:BT0oth:FILTer:PARAmeter:LPASs on page 190

### 4.1.2 Modulation settings

Access:

- ▶ Select "Baseband > Bluetooth > General > Filter/Clipping > Modulation Settings".



The tab provides settings to configure modulation settings.

#### Settings:

Modulation Type.....	93
Frequency Deviation.....	93
Modulation Index.....	94
Symbol Rate Variation.....	94

#### Modulation Type

Displays the modulation type used for the current packet selection.

Remote command:

[ :SOURce<hw> ] :BB:BT0oth:FILTer:MTYPE on page 188

#### Frequency Deviation

Enter the frequency deviation of the frequency modulated part.

The frequency deviation can be varied in a range from 100.0 kHz to 200.0 kHz according to Bluetooth specification.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:MSEttings:FDEVIation](#) on page 189

### Modulation Index

Displays the modulation index resulting from the entered frequency deviation value.

Modulation index is calculated from the given frequency deviation and symbol rate values.

The modulation index  $h$  is defined as:

$$h = \frac{2\Delta f}{f_{symbol}}$$

Where  $f_{symbol}$  is the "symbol rate" and  $\Delta f$  is the "frequency deviation".

According to the Bluetooth specification, the modulation index is allowed to vary between 0.28 and 0.35.

Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:FILTer:MINdex](#) on page 188

### Symbol Rate Variation

Enter the symbol rate.

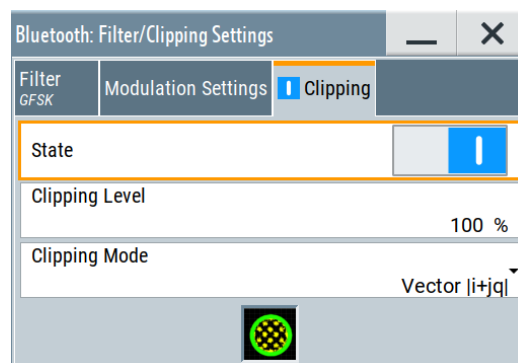
Remote command:

[\[:SOURCE<hw>\]:BB:BT0oth:SRATE:VARIation](#) on page 191

## 4.1.3 Clipping settings

Access:

- ▶ Select "Baseband > Bluetooth > General > Filter/Clipping > Clipping".



The dialog provides settings to configure clipping.

**Settings:**

Clipping State.....	95
Clipping Level.....	95
Clipping Mode.....	95

**Clipping State**

Switches baseband clipping on and off.

Baseband clipping is a simple and effective way of reducing the crest factor of the signal. Since clipping is done before to filtering, the procedure does not influence the spectrum. The EVM however increases.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:CLIPping:STATe` on page 186

**Clipping Level**

Sets the limit for clipping.

This value indicates at what point the signal is clipped. It is specified as a percentage, relative to the highest level. 100% indicates that clipping does not take place.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:CLIPping:LEVel` on page 186

**Clipping Mode**

Selects the clipping method. The dialog displays a graphical illustration on how this two methods work.

- "Vector  $| i + jq |$ "  
The limit is related to the amplitude  $| i + q |$ . The I and Q components are mapped together, the angle is retained.
- "Scalar  $| i | , | q |$ "  
The limit is related to the absolute maximum of all the I and Q values  $| i | + | q |$ . The I and Q components are mapped separately, the angle changes.

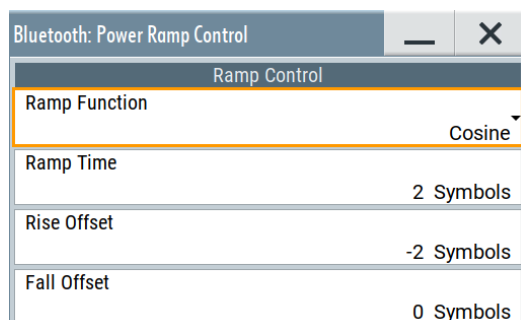
Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:CLIPping:MODE` on page 186

## 4.2 Power ramping settings

Access:

- ▶ Select "Bluetooth > General > Power Ramping".



The dialog provides settings to configure power ramping.

### Settings:

Ramp Function.....	96
Ramp Time.....	96
Rise Offset.....	96
Fall Offset.....	97

### Ramp Function

Selects the form of the transmitted power, i.e. the shape of the rising and falling edges during power ramp control.

- "Linear" The transmitted power rises and falls with linear fashion.
- "Cosine" The transmitted power rises and falls with a cosine-shaped edge. This setting causes a more favorable spectrum than the "Linear" setting.

Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:PRAMping:RFUNction](#) on page 192

### Ramp Time

Sets the ramp time, which extends the burst by a corresponding number of 0 padding symbols at the beginning and the end of a burst. During this period of time, power ramping is based on the specified ramp function.

Do not switch the transmitted power abruptly at the end or the start of a burst, since the switching operation generates excessively strong non-harmonics. The switching operation is therefore stretched over several symbol clocks.

Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:PRAMping:RTIME](#) on page 193

### Rise Offset

Sets the offset of the rising edge of a burst. The offset is specified by the selected number of symbols.



Negative values shift the rising edge to earlier positions, which results in a corresponding number of added 0 padding symbols before the burst.

Positive values shift the rising edge to later positions, which results in a corresponding number of skipped symbols at the beginning of the burst.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:PRAMPing:ROFFset` on page 193

#### Fall Offset

Sets the offset of the falling edge of a burst. The offset is specified by the selected number of symbols.

Negative values shift the falling edge to earlier positions, which results in a corresponding number of skipped symbols at the end of the burst.

Positive values shift the falling edge to later positions, which results in a corresponding number of added 0 padding symbols following the burst.

Remote command:

`[ :SOURCE<hw> ] :BB:BT0oth:PRAMPing:FOFFset` on page 192

## 4.3 Trigger settings

Access:

- ▶ Select "Baseband" > "Bluetooth" > "Trigger In".



This tab provides settings to select and configure the trigger, like trigger source, trigger mode and trigger delays, and to arm or trigger an internal trigger manually. The header of the tab displays the status of the trigger signal and trigger mode. As in the tabs "Marker" and "Clock", this tab provides also access to the settings of the related connectors.

#### Routing and activating a trigger signal

1. Define the effect of a trigger event and the trigger signal source.
  - a) Select "Trigger In" > "Mode".
  - b) Select "Trigger In" > "Source".
2. For external trigger signals, define the connector for signal input. See [Chapter 4.6, "Local and global connectors settings"](#), on page 105.

You can map trigger signals to one or more User x or T/M connectors.

Local and global connectors settings allow you to configure the signal mapping, the polarity, the trigger threshold and the input impedance of the input connectors.

3. Activate baseband signal generation. In the block diagram, set "Baseband" > "On".  
The R&S SMM100A starts baseband signal generation after the configured trigger event.

### About baseband trigger signals

This section focuses on the available settings.

For information on how these settings affect the signal, refer to section "Basics on ..." in the R&S SMM100A user manual.

### Settings:

Mode.....	98
Signal Duration Unit.....	99
Signal Duration.....	99
Running/Stopped.....	99
Time Based Trigger.....	99
Trigger Time.....	99
Arm.....	100
Execute Trigger.....	100
Source.....	100
Sync. Output to Ext. Trigger/Sync. Output to Trigger.....	100
External Inhibit/Trigger Inhibit.....	101
External Delay/Trigger Delay.....	101

### Mode

Selects trigger mode, i.e. determines the effect of a trigger event on the signal generation.

- "Auto"  
The signal is generated continuously.
- "Retrigger"  
The signal is generated continuously. A trigger event (internal or external) causes a restart.
- "Armed Auto"  
The signal is generated only when a trigger event occurs. Then the signal is generated continuously.  
An "Arm" stops the signal generation. A subsequent trigger event (internal or external) causes a restart.
- "Armed Retrigger"  
The signal is generated only when a trigger event occurs. Then the signal is generated continuously. Every subsequent trigger event causes a restart.  
An "Arm" stops signal generation. A subsequent trigger event (internal or external) causes a restart.
- "Single"  
The signal is generated only when a trigger event occurs. Then the signal is generated once to the length specified at "Signal Duration".  
Every subsequent trigger event (internal or external) causes a restart.

Remote command:

[ :SOURce<hw> ] :BB:BT0oth [ :TRIGger ] :SEQuence on page 195

**Signal Duration Unit**

Defines the unit for describing the length of the signal sequence to be output in the "Single" trigger mode.

Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:TRIGger:SLUNit](#) on page 197

**Signal Duration**

Requires trigger "Mode" > "Single".

Enters the length of the trigger signal sequence.

Use this parameter, for example, for the following applications:

- To output the trigger signal partly.
- To output a predefined sequence of the trigger signal.

Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:TRIGger:SLENgth](#) on page 197

**Running/Stopped**

With enabled modulation, displays the status of signal generation for all trigger modes.

- "Running"  
The signal is generated; a trigger was (internally or externally) initiated in triggered mode.
- "Stopped"  
The signal is not generated and the instrument waits for a trigger event.

Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:TRIGger:RMODe](#) on page 196

**Time Based Trigger**

Requires trigger "Mode" > "Armed Auto"/"Single".

Activates time-based triggering with a fixed time reference.

The R&S SMM100A triggers signal generation when its operating system time ("Current Time") matches a specified time trigger ("Trigger Time"). As trigger source, you can use an internal trigger or an external global trigger.

How to: Chapter "Time-based triggering" in the R&S SMM100A user manual.

Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:TRIGger:TIME\[:STATe\]](#) on page 197

**Trigger Time**

Requires trigger "Mode" > "Armed Auto"/"Single".

Sets date and time for a time-based trigger signal.

Set a trigger time that is later than the "Current Time". The current time is the operating system time of the R&S SMM100A. If you set an earlier trigger time than the current time, time-based triggering is not possible.

How to: Chapter "Time-based triggering" in the R&S SMM100A user manual.

"Date" Sets the date of the time-based trigger in format YYYY-MM-DD.

Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:TRIGger:TIME:DATE](#) on page 196

"Time" Sets the time of the time-based trigger in format hh:mm:ss.

Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:TRIGger:TIME:TIME](#) on page 196

### Arm

Stops the signal generation until subsequent trigger event occurs.

Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:TRIGger:ARM:EXECute](#) on page 198

### Execute Trigger

For internal trigger source, executes trigger manually.

Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:TRIGger:EXECute](#) on page 198

### Source

The following sources of the trigger signal are available:

- "Internal"  
The trigger event is executed manually by the "Execute Trigger".
- "External Global Trigger"  
The trigger event is the active edge of an external trigger signal provided and configured at the User x connectors.
- "Baseband Sync In"  
In primary-secondary instrument mode, secondary instruments are triggered by the active edge of the synchronization signal.

How to: ["Routing and activating a trigger signal"](#) on page 97

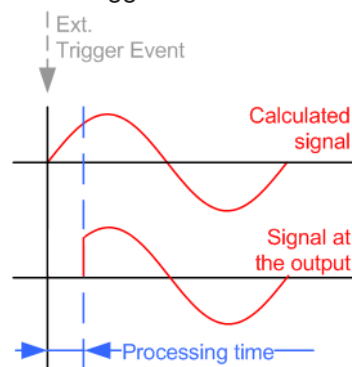
Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:TRIGger:SOURce](#) on page 195

### Sync. Output to Ext. Trigger/Sync. Output to Trigger

Enables signal output synchronous to the trigger event.

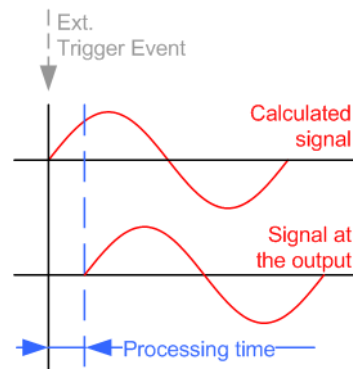
- "On"  
Corresponds to the default state of this parameter.  
The signal calculation starts simultaneously with the trigger event. Because of the processing time of the instrument, the first samples are cut off and no signal is output. After elapsing of the internal processing time, the output signal is synchronous to the trigger event.



- "Off"

The signal output begins after elapsing of the processing time. Signal output starts with sample 0. The complete signal is output.

This mode is recommended for triggering of short signal sequences. Short sequences are sequences with signal duration comparable with the processing time of the instrument.



Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger [ :EXTernal ] :SYNChronize:OUTPut`  
on page 198

#### External Inhibit/Trigger Inhibit

Applies for external trigger signal.

Sets the duration with that any following trigger event is suppressed. In "Retrigger" mode, for example, a new trigger event does not cause a restart of the signal generation until the specified inhibit duration does not expire.

For more information, see chapter "Basics" in the R&S SMM100A user manual.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger [ :EXTernal ] :INHibit` on page 199

#### External Delay/Trigger Delay

Delays the trigger event of the signal from:

- The external trigger source

Use this setting to:

- Synchronize the instrument with the device under test (DUT) or other external devices

For more information, see chapter "Basics on ..." in the R&S SMM100A user manual.

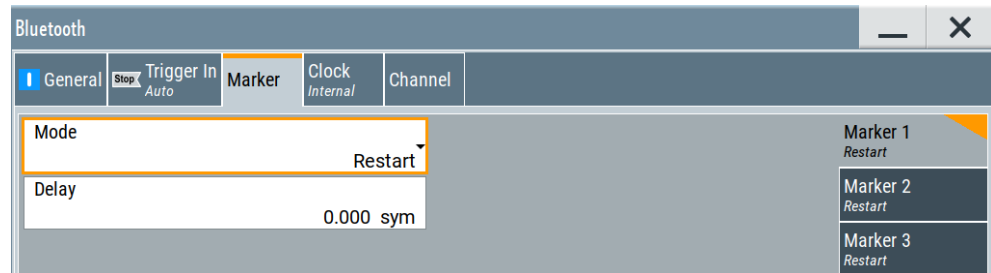
Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger [ :EXTernal ] :DELay` on page 198

## 4.4 Marker settings

Access:

- ▶ Select "Baseband" > "Bluetooth" > "Marker".



This tab provides settings to select and configure the marker output signal including marker mode and marker delay.

### Routing and activating a marker signal

1. To define the signal shape of an individual marker signal "x", select "Marker" > "Marker x" > "Mode".
2. Optionally, define the connector for signal output. See [Chapter 4.6, "Local and global connectors settings"](#), on page 105.  
You can map marker signals to one or more User x or T/M connectors.
3. Activate baseband signal generation. In the block diagram, set "Baseband" > "On".  
The R&S SMM100A adds the marker signal to the baseband signal. Also, R&S SMM100A outputs this signal at the configured User x connector.

### About marker output signals

This section focuses on the available settings.

For information on how these settings affect the signal, refer to section "Basics on ..." in the R&S SMM100A user manual.

### Settings:

Mode.....	102
Delay.....	104

### Mode

Marker configuration for up to 3 markers. The settings are used to select the marker mode defining the shape and periodicity of the markers. The contents of the dialog change with the selected marker mode.

How to: ["Routing and activating a marker signal"](#) on page 102

"Restart"            A marker signal is generated at the start of each signal sequence.

**"Event/Frame Start"**

A marker signal is generated at the start of each event/frame. The term event corresponds to a Bluetooth LE event, the term frame corresponds to a Bluetooth BR/EDR frame.

**"Event/Frame Active Part"/"Event/Frame Inactive Part"**

The marker masks the active/inactive part of the event/frame. At the start of each burst, the marker signal changes to high/low. It changes back to low/high after the end of each burst.

Shift the marker signal at the start/end of each burst with the parameters "Rising/Falling Edge Shift".

Also, for Bluetooth LE data packets higher than one, configure the "Packet Index". The index corresponds to the transmitted Tx event during the connection interval.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:FESHift`  
on page 200

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:RESHift`  
on page 203

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:PINdex`  
on page 202

**"Pulse"**

A regular marker signal is generated. The pulse frequency is defined by entering a divider. The frequency is derived by dividing the sample rate by the divider. The input box for the divider opens when "Pulse" is selected, and the resulting pulse frequency is displayed below it.

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:PULSe:DIVider` on page 202

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:PULSe:FREQuency?` on page 202

**"Pattern"**

A marker signal that is defined by a bit pattern is generated. The pattern has a maximum length of 32 bits and is defined in an input field which opens when pattern is selected.

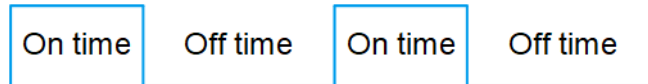
Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:PATTern`  
on page 202

**"On/Off Ratio"**

A regular marker signal that is defined by an On/Off ratio is generated. A period lasts one On and Off cycle.

If "Marker Mode > On/Off Ratio", specify the "On Time" and "Off Time", which are expressed as number of samples.



Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:ONTime`  
on page 201

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:OFFTime`  
on page 201

Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:MODE` on page 201

**Delay**

Delays the marker signal at the marker output relative to the signal generation start.

Variation of the parameter "Marker x" > "Delay" causes signal recalculation.

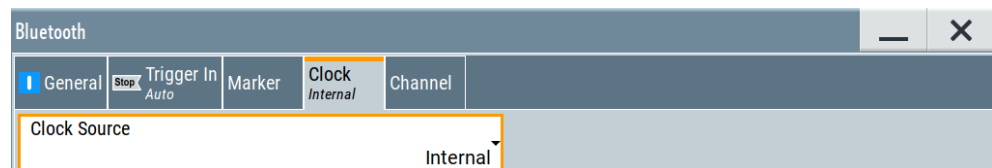
Remote command:

`[ :SOURce<hw> ] :BB:BT0oth:TRIGger:OUTPut<ch>:DELay` on page 200

## 4.5 Clock settings

Access:

- ▶ Select "Baseband" > "Bluetooth" > "Clock".



This tab provides settings to select and configure the clock signal, like the clock source and clock mode.

**Defining the clock**

1. Select "Clock" > "Source" to define the source of clock signal.
2. For external clock signals, define the connector for signal input. See [Chapter 4.6, "Local and global connectors settings"](#), on page 105.

You can map clock signals to one or more User x or T/M connectors.

Local and global connectors settings allow you to configure the signal mapping, the polarity, the trigger threshold and the input impedance of the input connectors.



3. Activate baseband signal generation. In the block diagram, set "Baseband" > "On".  
The R&S SMM100A starts baseband signal generation with a symbol rate that equals the clock rate.

### About clock signals

This section focuses on the available settings.

For information on how these settings affect the signal, refer to section "Basics on ..." in the R&S SMM100A user manual.

### Settings:

[Clock Source](#)..... 105

#### Clock Source

Selects the clock source.

- "Internal"  
The instrument uses its internal clock reference.

How to: "[Defining the clock](#)" on page 104

Remote command:

[\[:SOURce<hw>\]:BB:BT0oth:CLOCK:SOURce](#) on page 204

## 4.6 Local and global connectors settings

Accesses a dialog to configure local connectors or global connectors.

The button is available in the following dialogs or tabs:

- "Trigger / Marker / Clock" dialog that is accessible via the "TMC" block in the block diagram.
- "Trigger In", "Marker" and "Clock" tabs that are accessible via the "Baseband" block in the block diagram.



See also chapter "Local and global connectors settings" in the user manual.

## 5 Remote-control commands

The following commands are required to perform signal generation with the Bluetooth options in a remote environment. We assume that the R&S SMM100A has already been set up for remote operation in a network as described in the R&S SMM100A documentation. A knowledge about the remote control operation and the SCPI command syntax is assumed.



### Conventions used in SCPI command descriptions

For a description of the conventions used in the remote command descriptions, see section "Remote Control Commands" in the R&S SMM100A user manual.

### Common suffixes

The following common suffixes are used in remote commands:

Suffix	Value range	Description
SOURce<hw>	1	Available baseband signals
OUTPut<ch>	1 to 3	Available marker signals

### Programming examples

This description provides simple programming examples. The purpose of the examples is to present **all** commands for a given task. In real applications, one would rather reduce the examples to an appropriate subset of commands.

The programming examples have been tested with a software tool which provides an environment for the development and execution of remote tests. To keep the example as simple as possible, only the "clean" SCPI syntax elements are reported. Non-executable command lines (e.g. comments) start with two // characters.

At the beginning of the most remote control program, an instrument preset/reset is recommended to set the instrument to a definite state. The commands `*RST` and `SYSTem:PRESet` are equivalent for this purpose. `*CLS` also resets the status registers and clears the output buffer.

The following chapters describe the commands specific to the Bluetooth options R&S SMM-K60/R&S SMM-K117.

• <a href="#">General commands</a> .....	107
• <a href="#">Dirty transmitter configuration</a> .....	112
• <a href="#">Channel configuration commands - BR/EDR</a> .....	119
• <a href="#">Packet configuration commands - BR/EDR</a> .....	120
• <a href="#">Channel configuration commands - LE</a> .....	129
• <a href="#">Event and frame configuration commands - LE</a> .....	134
• <a href="#">Packet configuration commands - LE</a> .....	144
• <a href="#">Test packet configuration commands - LE</a> .....	183
• <a href="#">Filter/clipping commands</a> .....	185
• <a href="#">Power ramping commands</a> .....	192

- [Trigger commands](#)..... 194
- [Marker commands](#)..... 199
- [Clock commands](#).....203

## 5.1 General commands

### Example: To configure time units for timing and delay commands

The default time unit of timing and delay commands is millisecond. You can set time unit to millisecond or to second.

```
SOURcel:BB:BT0oth:UNIT:TIME?
// Response: "MS"
// Change time unit to seconds.
SOURcel:BB:BT0oth:UNIT:TIME S
```

### Example: To save and recall settings

```
MMEM:CDIR "/var/user/"
SOURcel:BB:BT0oth:PRESet
SOURcel:BB:BT0oth:SETTing:STORe "/var/user/Bluetooth_EDR"
SOURcel:BB:BT0oth:SETTing:CATalog?
// Response: Bluetooth_EDR,Bluetooth_SCO,BTO_test
SOURcel:BB:BT0oth:SETTing:DELeTe "BTO_test"
SOURcel:BB:BT0oth:SETTing:LOAD "BTO_dl"
SOURcel:BB:BT0oth:SETTing:CATalog?
// Response: "Bluetooth_EDR,Bluetooth_SCO"
```

### Example: To configure general Bluetooth BR/EDR settings

```
// *****
// Set time unit to ms.
// *****
SOURcel:BB:BT0oth:UNIT:TIME MS
// *****
// Set frequency and level.
// Set BR/EDR PHY and transport modes.
// Query the version of the digital standard.
// Generate and save a waveform file in the current directory.
// *****
SOURcel:FREQuency:CW 240200000
SOURcel:POWer:POWer -50
SOURcel:BB:BT0oth:BMODE BAS
SOURcel:BB:BT0oth:TMODE ACL
// SOURcel:BB:BT0oth:TMODE SCO
// SOURcel:BB:BT0oth:TMODE ESCO
SOURcel:BB:BT0oth:VERSion?
// Bluetooth version currently supported by the firmware, e.g. "5.4".
SOURcel:BB:BT0oth:STATe ON
SOURcel:BB:BT0oth:WAVEform:CREate "Bluetooth_EDR"
```

**Example: To configure general Bluetooth LE settings**

```
// *****
// Set time unit to ms.
// *****
SOURCE1:BB:BT00th:UNIT:TIME MS
// *****
// Set frequency and level.
// Set LE PHY and query the version of the digital standard.
// Generate and save a waveform file in the current directory.
// *****
SOURCE1:FREQuency:CW 240200000
SOURCE1:POWer:POWer -10
SOURCE1:BB:BT00th:BM0De BLEN
SOURCE1:BB:BT00th:VERSIon?
// Bluetooth version currently supported by the firmware, e.g. "5.4".
SOURCE1:BB:BT00th:STATe ON
SOURCE1:BB:BT00th:WAVEform:CREate "Bluetooth_LE"
```

[:SOURCE<hw>]:BB:BT00th:BCText?.....	108
[:SOURCE<hw>]:BB:BT00th:BM0De.....	109
[:SOURCE<hw>]:BB:BT00th:PRESet.....	109
[:SOURCE<hw>]:BB:BT00th:SETTing:CATalog.....	109
[:SOURCE<hw>]:BB:BT00th:SETTing:DELeTe.....	110
[:SOURCE<hw>]:BB:BT00th:SETTing:LOAD.....	110
[:SOURCE<hw>]:BB:BT00th:SETTing:STORE.....	110
[:SOURCE<hw>]:BB:BT00th:STATe.....	110
[:SOURCE<hw>]:BB:BT00th:TM0De.....	111
[:SOURCE<hw>]:BB:BT00th:UNIT:TIME.....	111
[:SOURCE<hw>]:BB:BT00th:VERSIon?.....	111
[:SOURCE<hw>]:BB:BT00th:WAVEform:CREate.....	112

**[:SOURCE<hw>]:BB:BT00th:BCText?**

Queries the state and controller role.

**Return values:**

<BcText>                    string

**Connected**  
                              (only data channel type)

**Advertiser**  
                              (only advertising channel type)  
                              ADV\_IND, ADV\_DIRECT\_IND, ADV\_NONCONN\_IND,  
                              ADV\_SCAN\_IND  
                              Within R&S SMM-K117 also ADV\_EXT\_IND, AUX\_ADV\_IND,  
                              AUX\_SYNC\_IND, AUX\_CHAIN\_IND

**Scanner**

(only advertising channel type)

SCAN\_REQ, SCAN\_RSP

Within R&S SMM-K117 also AUX\_SCAN\_REQ,  
AUX\_SCAN\_RSP**Initiator**

(only advertising channel type)

CONNECT\_IND

Within R&S SMM-K117 also AUX\_CONNECT\_REQ,  
AUX\_CONNECT\_RSP

- Example:** See [Example "To configure general Bluetooth BR/EDR settings"](#) on page 107.
- Usage:** Query only
- Manual operation:** See ["Bluetooth Controller State"](#) on page 53

**[ :SOURCE<hw> ]:BB:BT0oth:BM0De <BMode>**

Determines the Bluetooth mode.

**Parameters:**

&lt;BMode&gt;                   BASic | BLENergy

**BASic**

Selects Bluetooth mode BR + EDR.

**BLENergy**

Selects Bluetooth LE.

\*RST:           BASic

- Example:** See [Example "To configure general Bluetooth BR/EDR settings"](#) on page 107.
- Manual operation:** See ["Bluetooth Mode"](#) on page 34

**[ :SOURCE<hw> ]:BB:BT0oth:PRESet**

Sets the parameters of the digital standard to their default values (\*RST values specified for the commands).

Not affected is the state set with the command SOURCE&lt;hw&gt;:BB:BT0oth:STATe.

- Example:** See [Example "To save and recall settings"](#) on page 107.
- Usage:** Event
- Manual operation:** See ["Set To Default"](#) on page 33

**[ :SOURCE<hw> ]:BB:BT0oth:SETTING:CATalog <Catalog>**

Queries the files with settings in the default directory. Listed are files with the file extension \*.bto.

**Parameters:**

<Catalog> string  
Returns a string of file names separated by commas.

**Example:** See [Example "To save and recall settings"](#) on page 107.

**Manual operation:** See ["Save/Recall"](#) on page 34

**[:SOURce<hw>]:BB:BT0oth:SETTing:DELeTe <Filename>**

Deletes the selected file from the default or specified directory. Deleted are files with the file extension \*.bto.

**Parameters:**

<Filename> string  
file name or complete file path; file extension can be omitted

**Example:** See [Example "To save and recall settings"](#) on page 107.

**Manual operation:** See ["Save/Recall"](#) on page 34

**[:SOURce<hw>]:BB:BT0oth:SETTing:LOAD <Filename>**

Loads the selected file from the default or the specified directory. Loaded are files with extension \*.bto.

**Parameters:**

<Filename> string  
file name or complete file path; file extension can be omitted

**Example:** See [Example "To save and recall settings"](#) on page 107.

**Manual operation:** See ["Save/Recall"](#) on page 34

**[:SOURce<hw>]:BB:BT0oth:SETTing:STORe <Filename>**

Saves the current settings into the selected file; the file extension (\*.bto) is assigned automatically.

**Setting parameters:**

<Filename> string  
file name or complete file path

**Example:** See [Example "To save and recall settings"](#) on page 107.

**Usage:** Setting only

**Manual operation:** See ["Save/Recall"](#) on page 34

**[:SOURce<hw>]:BB:BT0oth:STATe <State>**

Activates the standard and deactivates all the other digital standards and digital modulation modes in the same path.

**Parameters:**

<State> 1 | ON | 0 | OFF  
 \*RST: 0

**Example:** See [Example "To configure general Bluetooth BR/EDR settings"](#) on page 107.

**Manual operation:** See ["State"](#) on page 33

**[[:SOURce<hw>]:BB:BT0oth:TMODe <TMode>**

Selects the transport mode.

**Parameters:**

<TMode> ACL | SCO | ESCO

**ACL**

Asynchronous connection-less mode used for a point-to-point multipoint link between a Central and all Peripherals.

**SCO**

Synchronous connection-oriented mode used for a point-to-point link between a Central and a specific Peripheral.

**ESCO**

Enhanced synchronous connection-oriented mode used for a symmetric or asymmetric point-to-point link between a Central and a specific Peripheral.

\*RST: ACL

**Example:** See [Example "To configure general Bluetooth BR/EDR settings"](#) on page 107.

**Manual operation:** See ["Transport Mode"](#) on page 34

**[[:SOURce<hw>]:BB:BT0oth:UNIT:TIME <Time>**

Sets the time unit for remote control commands.

**Parameters:**

<Time> S | MS  
 \*RST: MS

**Example:** See [Example "To configure time units for timing and delay commands"](#) on page 107.

**[[:SOURce<hw>]:BB:BT0oth:VERSion?**

Queries the version of the specification for Bluetooth wireless technology underlying the definitions.

**Return values:**

<Version> string

<b>Example:</b>	See <a href="#">Example "To configure general Bluetooth BR/EDR settings"</a> on page 107.
<b>Usage:</b>	Query only
<b>Options:</b>	Version 5.x requires R&S SMM-K117.
<b>Manual operation:</b>	See <a href="#">"Bluetooth Version"</a> on page 34

**[:SOURce<hw>]:BB:BT0oth:WAVeform:CREate <Filename>**

Saves the current settings as an ARB signal in a waveform file (\*.wv).

**Setting parameters:**

<Filename> string

**Example:** See [Example "To configure general Bluetooth BR/EDR settings"](#) on page 107.

**Usage:** Setting only

**Manual operation:** See ["Generate Waveform"](#) on page 34

## 5.2 Dirty transmitter configuration

**Example: To configure a dirty transmitter**

```
// *****
// Set modulation index mode to stable.
// *****
SOURcel:BB:BT0oth:DTTest:MIMode STAB

// *****
// Reset dirty transmitter. Set frequency drift rate, start phase,
// frequency drift deviation and number of packets
// per dirty transmitter set (LE only). Enable dirty transmitter.
// *****
SOURcel:BB:BT0oth:DTTest:STDefault
SOURcel:BB:BT0oth:DTTest:FDRate 1.25
SOURcel:BB:BT0oth:DTTest:SPHase 0
SOURcel:BB:BT0oth:DTTest:FDDeviation 50
SOURcel:BB:BT0oth:DTTest:NPPSet NP50

// *****
// Enable all long sets for LE or BR dirty transmitter.
// *****
SOURcel:BB:BT0oth:DTTest:TABLE:LONG:SET1:STATE 1
SOURcel:BB:BT0oth:DTTest:TABLE:LONG:SET2:STATE 1
SOURcel:BB:BT0oth:DTTest:TABLE:LONG:SET3:STATE 1
SOURcel:BB:BT0oth:DTTest:TABLE:LONG:SET4:STATE 1
SOURcel:BB:BT0oth:DTTest:TABLE:LONG:SET5:STATE 1
```



```
SOURce1:BB:BT0oth:DTTest:TABLE:LONG:SET6:STATE 1
SOURce1:BB:BT0oth:DTTest:TABLE:LONG:SET7:STATE 1
SOURce1:BB:BT0oth:DTTest:TABLE:LONG:SET8:STATE 1
SOURce1:BB:BT0oth:DTTest:TABLE:LONG:SET9:STATE 1
SOURce1:BB:BT0oth:DTTest:TABLE:LONG:SET10:STATE 1
```

```
// *****
// Enable dirty transmitter.
// *****
SOURce1:BB:BT0oth:DTTest:DTTState 1
```

<code>[SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:DTTState</code> .....	113
<code>[SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:FDDeviation</code> .....	113
<code>[SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:FDRate</code> .....	114
<code>[SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:MIIMode</code> .....	114
<code>[SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:NPPSet</code> .....	114
<code>[SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:SPHase</code> .....	115
<code>[SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:STDefault</code> .....	115
<code>[SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:TABLE</code> .....	115
<code>[SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:TABLE:LONG:SET&lt;ch&gt;:CFOffset</code> .....	115
<code>[SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:TABLE:LONG:SET&lt;ch&gt;:MINDex</code> .....	116
<code>[SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:TABLE:LONG:SET&lt;ch&gt;:STATE</code> .....	117
<code>[SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:TABLE:LONG:SET&lt;ch&gt;:STERror</code> .....	117
<code>[SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:TABLE:SHORT:SET&lt;ch&gt;:CFOffset</code> .....	117
<code>[SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:TABLE:SHORT:SET&lt;ch&gt;:STATE</code> .....	118
<code>[SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:TABLE:SHORT:SET&lt;ch&gt;:STERror</code> .....	118

---

#### `[SOURce<hw>]:BB:BT0oth:DTTest:DTTState <DttState>`

Activates the "Dirty Transmitter Test".

For EDR packets, the parameter sets apply for 20 packets each.

#### Parameters:

`<DttState>`            1 | ON | 0 | OFF  
                       \*RST:        0

**Example:**            See [Example "To configure a dirty transmitter"](#) on page 112.

**Manual operation:** See ["Dirty Transmitter Test"](#) on page 37

---

#### `[SOURce<hw>]:BB:BT0oth:DTTest:FDDeviation <FdDeviation>`

Sets a frequency drift rate.

A sine wave is used to drift the modulated Bluetooth signal around center frequency + carrier frequency offset. The maximum deviation reached during the drift equals the set frequency drift deviation.

#### Parameters:

`<FdDeviation>`        integer  
                       Range:        -100 to 100  
                       \*RST:        25

**Example:** See [Example "To configure a dirty transmitter"](#) on page 112.

**Manual operation:** See ["Frequency Drift Deviation \(+/-\)"](#) on page 38

**[:SOURce<hw>]:BB:BT0oth:DTTest:FDRate <FdRate>**

Sets a frequency drift rate.

A sine wave is used to drift the modulated Bluetooth signal around center frequency + carrier frequency offset with the set frequency drift rate.

**Parameters:**

<FdRate> 0.3 KHz | 0.5 KHz | 1.6 KHz | 10 KHz  
 Range: depends on packet type to depends on packet type  
 Increment: 0.001  
 \*RST: depends on packet type

**Example:** See [Example "To configure a dirty transmitter"](#) on page 112.

**Manual operation:** See ["Frequency Drift Rate"](#) on page 38

**[:SOURce<hw>]:BB:BT0oth:DTTest:MIMode <MIMode>**

Determines standard or stable mode for the modulation index of dirty transmitter according to the Bluetooth core specification.

**Parameters:**

<MIMode> STANdard | STABLE  
 \*RST: STANdard

**Example:** See [Example "To configure a dirty transmitter"](#) on page 112.

**Options:** R&S SMM-K117

**Manual operation:** See ["Modulation Index Mode"](#) on page 38

**[:SOURce<hw>]:BB:BT0oth:DTTest:NPPSet <NumPack>**

Specifies the number of packets per dirty transmitter set.

Bluetooth mode	Channel type	Number of packets
BR + EDR	-	50, 2, 1 packets
LE	Advertising, data	50, 2, 1 packets

**Parameters:**

<NumPack> NP50 | NP2 | NP1  
 \*RST: NP50

**Example:** See [Example "To configure a dirty transmitter"](#) on page 112.

**Manual operation:** See ["Number of Packets per Set"](#) on page 38

---

**[[:SOURce<hw>]:BB:BT0oth:DTTest:SPHase <SPHase>**

The command enters a start phase.

The start phase applies to the sine wave that is used to drift the modulated Bluetooth signal. The drift is around the center frequency plus the carrier frequency offset.

**Parameters:**

<SPHase>	integer
	Range: 0 to 359
	Increment: 1
	*RST: 0
	Default unit: degree

**Example:**           SOURce1:BB:BT0:DTT:SPH 0  
enters a start phase.

**Example:**           See also [Example "To configure a dirty transmitter"](#) on page 112.

**Manual operation:** See ["Start Phase"](#) on page 37

---

**[[:SOURce<hw>]:BB:BT0oth:DTTest:STDefault**

The command calls the default settings for the Dirty Transmitter Test.

**Example:**           SOURce1:BB:BT0:DTT:STD  
calls the default settings.

**Example:**           See also [Example "To configure a dirty transmitter"](#) on page 112.

**Usage:**             Event

**Manual operation:** See ["Set To Default"](#) on page 37

---

**[[:SOURce<hw>]:BB:BT0oth:DTTest:TABLE <Table>**

Opens the table settings.

**Parameters:**

<Table>	NOTable   SHORt   LONG
---------	------------------------

**Example:**           SOURce1:BB:BT0:PTYP DH1  
calls the default settings.

SOURce1:BB:BT0:DTT:TABL LONG

See also [Example "To configure a dirty transmitter"](#) on page 112.

**Example:**           calls the default settings.

---

**[[:SOURce<hw>]:BB:BT0oth:DTTest:TABLE:LONG:SET<ch>:CFOffset <CfOffset>**

Sets a carrier frequency offset.

The carrier frequency offset shows the deviation of the transmitted initial center frequency from carrier frequency.

**Parameters:**

<CfOffset> integer  
 Range: -150 to 150  
 \*RST: 1  
 Default unit: kHz

**Example:**

SOURce1:BB:BTO:PTYP DH1  
 sets the packet type.  
 SOURce1:BB:BTO:DTT:TABL LONG  
 sets the table type  
 SOURce1:BB:BTO:DTT:TABL:LONG:SET2:CFOF 14  
 sets a carrier frequency offset.

**Example:** See also [Example "To configure a dirty transmitter"](#) on page 112.

**Manual operation:** See ["Carrier Frequency Offset kHz"](#) on page 39

**[ :SOURce<hw>]:BB:BTOoth:DTTtest:TABLE:LONG:SET<ch>:MINDex <MIndex>**

Sets the modulation index, that specifies the frequency deviation.

The modulation index h is defined as:

$$h = \frac{2\Delta f}{f_{symbol}}$$

with

$f_{symbol}$  = "symbol rate", set with the command [\[:SOURce<hw>\]:BB:BTOoth:SRATE:VARIation](#)

$\Delta f$  = "frequency deviation", set with the command [\[:SOURce<hw>\]:BB:BTOoth:MSETtings:FDEVIation](#)

According to the Bluetooth standard, the modulation index is allowed to vary between 0.28 and 0.35.

**Parameters:**

<MIndex> float  
 Range: 0.28 to 0.55  
 Increment: 0.01  
 \*RST: 0.28

**Example:**

SOURce1:BB:BTO:PTYP DH1  
 sets the packet type.  
 SOURce1:BB:BTO:DTT:TABL LONG  
 enters the table type  
 SOURce1:BB:BTO:DTT:TABL:LONG:SET2:MIND 0.3  
 enters a modulation index.

**Example:** See also [Example "To configure a dirty transmitter"](#) on page 112.

**Manual operation:** See ["Modulation Index"](#) on page 39

---

```
[ :SOURce<hw>]:BB:BTOoth:DTTTest:TABLE:LONG:SET<ch>:STATe <State>
```

Activates the corresponding parameter set for the long table.

For basic rate packets, each set applies to 20 ms of signal.

**Parameters:**

```
<State>          1 | ON | 0 | OFF
*RST:           1
```

**Example:**

```
SOURce1:BB:BTO:PTYP DH1
sets the packet type.
SOURce1:BB:BTO:DTT:TABL LONG
sets the table type
SOURce1:BB:BTO:DTT:TABL:LONG:SET2:STAT ON
activates the set 2 in the long table.
```

**Example:** See also [Example "To configure a dirty transmitter"](#) on page 112.

**Manual operation:** See ["State"](#) on page 38

---

```
[ :SOURce<hw>]:BB:BTOoth:DTTTest:TABLE:LONG:SET<ch>:STERror <StError>
```

Sets a symbol timing error in ppm.

This parameter modifies the symbol clock frequency by the set value.

**Parameters:**

```
<StError>       integer
Range:          -150 to 150
*RST:           1
Default unit:   ppm
```

**Example:**

```
SOURce1:BB:BTO:PTYP DH1
sets the packet type.
SOURce1:BB:BTO:DTT:TABL LONG
sets the table type
SOURce1:BB:BTO:DTT:TABL:LONG:SET2:STER -20
sets a symbol timing error.
```

**Example:** See also [Example "To configure a dirty transmitter"](#) on page 112.

**Manual operation:** See ["Symbol Timing Error"](#) on page 39

---

```
[ :SOURce<hw>]:BB:BTOoth:DTTTest:TABLE:SHORT:SET<ch>:CFOFfset <CfOffset>
```

Sets a carrier frequency offset.

The carrier frequency offset shows the deviation of the transmitted initial center frequency from carrier frequency.

**Parameters:**

<CfOffset> integer  
 Range: -150 to 150  
 \*RST: 1  
 Default unit: kHz

**Example:**

SOURce1:BB:BTO:PTYP DH1  
 sets the packet type.  
 SOURce1:BB:BTO:DTT:TABL SHOR  
 sets the table type  
 SOURce1:BB:BTO:DTT:TABL:SHOR:SET2:CFOF 65  
 sets a carrier frequency offset.

**Example:** See also [Example "To configure a dirty transmitter"](#) on page 112.

**Manual operation:** See ["Carrier Frequency Offset kHz"](#) on page 39

**[:SOURce<hw>]:BB:BTOoth:DTTtest:TABLE:SHORT:SET<ch>:STATE <State>**

Activates the corresponding parameter set in the short table. If a set deactivated, its parameters are skipped in the sequence. Instead, the next active set is used.

For EDR packets, each set applies to 20 packets.

**Parameters:**

<State> 1 | ON | 0 | OFF  
 \*RST: 1

**Example:**

SOURce1:BB:BTO:PTYP DH1  
 sets the packet type.  
 SOURce1:BB:BTO:DTT:TABL SHOR  
 sets the table type  
 SOURce1:BB:BTO:DTT:TABL:SHOR:SET2:STAT ON  
 activates the set 2 in the short table.

**Example:** See also [Example "To configure a dirty transmitter"](#) on page 112.

**Manual operation:** See ["State"](#) on page 38

**[:SOURce<hw>]:BB:BTOoth:DTTtest:TABLE:SHORT:SET<ch>:STERror <StError>**

Sets a symbol timing error in ppm.

The symbol timing error modifies the symbol clock frequency by the set amount.

**Parameters:**

<StError> integer  
 Range: -150 to 150  
 Increment: 1  
 \*RST: 1

**Example:**            `SOURce1:BB:BTO:PTYP DH1`  
                          sets the packet type.  
                          `SOURce1:BB:BTO:DTT:TABL SHOR`  
                          enters the table type  
                          `SOURce1:BB:BTO:DTT:TABL:SHOR:SET2:STER 20`  
                          enters a symbol timing error.

**Example:**            See also [Example "To configure a dirty transmitter"](#) on page 112.

**Manual operation:** See ["Symbol Timing Error"](#) on page 39

## 5.3 Channel configuration commands - BR/EDR

<a href="#">[:SOURce&lt;hw&gt;]:BB:BTOoth:PTYPe</a> .....	119
<a href="#">[:SOURce&lt;hw&gt;]:BB:BTOoth:SLENgth</a> .....	119
<a href="#">[:SOURce&lt;hw&gt;]:BB:BTOoth:STIMing</a> .....	120

---

### **[:SOURce<hw>]:BB:BTOoth:PTYPe <PType>**

The available packets depend on the selected transport mode. All packet types as defined in the Bluetooth specifications are supported.

**Parameters:**

<PType>            ID | NULL | POLL | FHS | DM1 | DH1 | DM3 | DH3 | DM5 | DH5 |  
 AUX1 | ADH1 | ADH3 | ADH5 | AEDH1 | AEDH3 | AEDH5 |  
 HV1 | HV2 | HV3 | DV | EV3 | EV4 | EV5 | EEV3 | EEV5 |  
 EEEV3 | EEEV5  
 \*RST:            DH1

**Example:**            `BB:BTO:PTYP NULL`  
                          Sets a null packet.

**Manual operation:** See ["Packet Type"](#) on page 40

---

### **[:SOURce<hw>]:BB:BTOoth:SLENgth <SLength>**

Sets the sequence length of the Bluetooth signal in number of frames. This signal is calculated in advance and output in the arbitrary waveform generator.

**Parameters:**

<SLength>            integer  
 Range:            depends on the number of states in dirty transmitter  
                          test to dynamic  
 \*RST:            1

**Example:**            `BB:BTO:SLEN 10`  
                          sets the sequence length to 10 frames.

**Manual operation:** See ["Sequence Length"](#) on page 40

---

**[:SOURce<hw>]:BB:BT0oth:STIMing <SlotTiming>**

Selects the Rx slot timing mode.

**Parameters:**

<SlotTiming> TX | LOOPback  
\*RST: TX

**Example:**

BB:BT0:PTYP DH3  
sets the packet type.  
BB:BT0:STIM LOOP  
selects loopback test mode.

**Manual operation:** See "Slot Timing" on page 41

## 5.4 Packet configuration commands - BR/EDR

<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:ACKNOWLEDgement.....</a>	120
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:BDALap.....</a>	121
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:BDANap.....</a>	121
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:BDAUap.....</a>	122
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:CODevice.....</a>	122
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:DATA.....</a>	122
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:DATA:DPATtern.....</a>	123
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:DATA:DSElection.....</a>	123
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:DATA:VDPAtern.....</a>	123
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:DATA:VDSElection.....</a>	124
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:DLEngth.....</a>	124
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:DSFPacket.....</a>	124
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:DWHitening.....</a>	125
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:EIRPacketfollows.....</a>	125
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:HFControl.....</a>	126
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:LFSWord.....</a>	126
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:LTAddress.....</a>	126
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:PFControl.....</a>	127
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:PLEngth.....</a>	127
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:SLAP.....</a>	128
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:SNSValue.....</a>	128
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:SRMode.....</a>	128
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:PCONfiguration:VDATA.....</a>	129

---

**[:SOURce<hw>]:BB:BT0oth:PCONfiguration:ACKNOWLEDgement  
<Acknowledgement>**

Sets the ARQN bit of the packet header..

**Parameters:**

<Acknowledgement> NAK | ACK



**NAK**

Request to retransmit the previous payload.

**ACK**

Previous payload has been received successfully.

\*RST: ACK

**Example:**

BB:BTO:PTYP DH1

selects the packet type DH1.

BB:BTO:PCON:DSFP PED

enable packet editor under data source for packet

BB:BTO:PCON:ACKN ACK

sets positive acknowledgement

**Manual operation:** See "[Acknowledgment](#)" on page 44

**[:SOURCE<hw>]:BB:BT0oth:PCONfiguration:BDALap <BdaLap>, <BitCount>**

Sets the lower address part of Bluetooth Device Address. The length of LAP is 24 bits or 6 hexadecimal figures.

**Parameters:**

<BdaLap>	numeric
	Range: #H000000 to #HFFFFFF
	*RST: 80
<BitCount>	integer
	Range: 8 to 24
	*RST: 24

**Example:** BB:BTO:PCON:BDAL #H000000,24  
Sets the lower address part.

**Manual operation:** See "[Bluetooth Device Address \(BD\\_ADDR\)](#)" on page 43

**[:SOURCE<hw>]:BB:BT0oth:PCONfiguration:BDANap <BdaNap>, <BitCount>**

Enters the non-significant address part of Bluetooth Device Address. The length of NAP is 16 bits or 4 hexadecimal figures.

**Parameters:**

<BdaNap>	numeric
	Range: #H0000 to #HFFFF
	*RST: ABCD
<BitCount>	integer
	Range: 16 to 16
	*RST: 16

**Example:** BB:BTO:PCON:BDAN #H0000,16  
Sets the non-significant address part.

**Manual operation:** See "[Bluetooth Device Address \(BD\\_ADDR\)](#)" on page 43

---

**[[:SOURce<hw>]:BB:BT0oth:PCONfiguration:BDAUap <BdaUap>, <BitCount>**

Enters the upper address part of Bluetooth Device Address. The length of UAP is 8 bits or 2 hexadecimal figures.

**Parameters:**

<BdaUap>	numeric
	Range: #H00 to #HFF
	Increment: 1
	*RST: 48
<BitCount>	integer
	Range: 8 to 8
	*RST: 8

**Example:** BB:BT0:PCON:BDAN #H00,8  
Sets the non-significant address part.

**Manual operation:** See "[Bluetooth Device Address \(BD\\_ADDR\)](#)" on page 43

---

**[[:SOURce<hw>]:BB:BT0oth:PCONfiguration:CODevice <CoDevice>, <BitCount>**

A parameter received during the device discovery procedure, indicates the type of device and which types of service that are supported.

**Parameters:**

<CoDevice>	numeric
	Range: #H000000 to #HFFFFFF
	*RST: #H0
<BitCount>	integer
	Range: 24 to 24
	*RST: 24

**Example:** BB:BT0:PTYP FHS  
Sets the packet type.  
BB:BT0:PCON:DSFP PED  
Enables the packet editor under data source for the packet.  
BB:BT0:PCON:COD #H020104,24  
Sets the class of device.

**Manual operation:** See "[Class of Device](#)" on page 47

---

**[[:SOURce<hw>]:BB:BT0oth:PCONfiguration:DATA <Data>**

Selects the data source used for the payload.

**Parameters:**

<Data>	ALL0   ALL1   PATtern   PN09   PN11   PN15   PN16   PN20   PN21   PN23   DLISt
	*RST: PN09

**Example:** BB:BTO:PTYP FHS  
sets the packet type  
BB:BTO:PCON:DSFP PED  
enable packet editor under data source for packet  
BB:BTO:PCON:DATA ALL1  
sets the data type.

**Manual operation:** See "[Data Source](#)" on page 45

**[:SOURCE<hw>]:BB:BTOoth:PCONfiguration:DATA:DPATtern <DPattern>, <BitCount>**

Selects the data for a pattern.

**Parameters:**

<DPattern> numeric  
\*RST: #H0  
  
<BitCount> integer  
Range: 1 to 64  
\*RST: 1

**Example:** BB:BTO:PCON:DATA PATT  
Sets the data type.  
BB:BTO:PCON:DATA:DPAT #B010101,6  
Selects the data for a pattern with the length of 6 bits.

**Manual operation:** See "[Data Source](#)" on page 45

**[:SOURCE<hw>]:BB:BTOoth:PCONfiguration:DATA:DSElection <DSelection>**

The command selects data list file.

**Parameters:**

<DSelection> string  
Increment: 1

**Example:** BB:BTO:PCON:DATA DLIS  
selects the data type.  
BB:BTO:PCON:DSEL bluetooth\_1  
selects the file for the data.

**Manual operation:** See "[Data Source](#)" on page 45

**[:SOURCE<hw>]:BB:BTOoth:PCONfiguration:DATA:VDPATtern <VdPattern>, <BitCount>**

Sets the bit pattern for the voice data.

**Parameters:**

<VdPattern> numeric  
\*RST: #H0

<BitCount> integer  
 Range: 1 to 64  
 \*RST: 1

**Example:** BB:BTO:PCON:DATA:PATT  
 Selects the data type.  
 BB:BTO:PCON:DATA:VDPA #B010101,6  
 Selects the bit pattern for the voice data with the length of 24 bits.

**Manual operation:** See ["Data Source \(Voice Field\)"](#) on page 47

**[:SOURCE<hw>]:BB:BTObth:PCONfiguration:DATA:VDSElection <VdSelection>**

Selects the data list for voice data.

**Parameters:**

<VdSelection> string

**Example:** BB:BTO:PCON:VDAT DLIS  
 selects the data type.  
 BB:BTO:PCON:VDSE bluetooth\_1  
 selects the file for the data.

**Manual operation:** See ["Data Source \(Voice Field\)"](#) on page 47

**[:SOURCE<hw>]:BB:BTObth:PCONfiguration:DLENgth <DLength>**

Sets the payload data length in bytes.

**Parameters:**

<DLength> integer  
 Range: 0 to depends on packet type  
 Increment: 1  
 \*RST: depends on packet type

**Example:** BB:BTO:PTYP DH1  
 sets the packet type.  
 BB:BTO:PCON:DSFP PED  
 enable packet editor under data source for packet  
 BB:BTO:PCON:DLEN 25  
 sets the data length.

**Manual operation:** See ["Data Length"](#) on page 46

**[:SOURCE<hw>]:BB:BTObth:PCONfiguration:DSFPacket <DsfPacket>**

Selects the data source for the selected packet type.

**Parameters:**

<DsfPacket> PEDit | ADATa

**PED**

Enables the "Packet Editor". All packet fields can be configured individually.

**ADAT**

Fills the generated packets with the selected data source. Useful if predefined data contents are loaded with a data list file or the data contents of the packet are not of interest.

\*RST: PEDit

**Example:**

BB:BTO:PCON:DSFP PED

enables packet editor under data source for packet.

**Manual operation:** See "[Data Source for Packet](#)" on page 43

**[:SOURCE<hw>]:BB:BTOoth:PCONfiguration:DWHitening <DWhitening>**

Activates the "Data Whitening".

**Parameters:**

<DWhitening> 1 | ON | 0 | OFF

\*RST: 0

**Example:**

BB:BTO:PCON:DWH ON

activates data whitening.

**Manual operation:** See "[Data Whitening](#)" on page 43

**[:SOURCE<hw>]:BB:BTOoth:PCONfiguration:EIRPacketfollows <EirPacketFollow>**

Indicates that an extended inquiry response packet can follow.

**Parameters:**

<EirPacketFollow> YES | NO

**YES**

Indicates that EIR packet follows.

**NO**

Indicates that EIR packet does not follow.

\*RST: NO

**Example:**

BB:BTO:PCON:PTYP FHS

sets the packet type.

BB:BTO:PCON:DSFP PED

enables the packet editor under data source for the packet

BB:BTO:PCON:EIRP YES

the EIR packet follows.

**Manual operation:** See "[EIR packet follows](#)" on page 46

**[[:SOURce<hw>]:BB:BT0oth:PCONfiguration:HFControl <HfControl>**

The command sets the FLOW bit in the header. This bit indicates start or stop of transmission of packets over the ACL logical transport.

**Parameters:**

<HfControl> GO | STOP

**GO**

Allows the other devices to transmit new data.

**STOP**

Stops the other devices from transmitting data temporarily.

\*RST: GO

**Example:**

```
BB:BT0:PCON:PTYP DH1
```

sets the packet type.

```
BB:BT0:PCON:DSFP PED
```

enable packet editor under data source for packet.

```
BB:BT0:PCON:HFC GO
```

allows the other devices to transmit new data.

**Manual operation:** See "[Flow Control](#)" on page 44

**[[:SOURce<hw>]:BB:BT0oth:PCONfiguration:LFSWord <LapForSW>, <BitCount>**

Sets the lower address part (LAP) of the sync word for FHS packets. The length of LAP is 24 bits or 6 hexadecimal figures.

**Parameters:**

<LapForSW> numeric

Range: #H000000 to #FFFFFFF

\*RST: #H000080

<BitCount> integer

Range: 8 to 24

\*RST: 24

**Example:**

```
BB:BT0:PCON:LFSW #H000080,24
```

Sets the lower address part.

**Manual operation:** See "[LAP for Sync Word](#)" on page 43

**[[:SOURce<hw>]:BB:BT0oth:PCONfiguration:LTAddress <LtAddress>**

The command enters the logical transport address for the header. Each Peripheral active in a piconet is assigned a primary logical transport address (LT\_ADDR). The all-zero LT\_ADDR is reserved for broadcast messages.

**Parameters:**

<LtAddress> integer  
 Range: 0 to 7  
 Increment: 1  
 \*RST: 0

**Example:**

BB:BTO:PCON:PTYP DH1  
 sets the packet type.  
 BB:BTO:PCON:DSFP PED  
 enable packet editor under data source for packet  
 BB:BTO:PCON:LTAD 0  
 sets the logical transport address equal zero.

**Manual operation:** See "[Logical Transport Address](#)" on page 44

**[:SOURCE<hw>]:BB:BTOoth:PCONfiguration:PFControl <PfControl>**

The command sets the FLOW bit in the payload (flow control per logical link).

**Parameters:**

<PfControl> GO | STOP  
**GO**  
 Indicates the start of transmission of ACL packets after a new connection has been established.  
**STOP**  
 Indicates the stop of transmission of ACL packets before an additional amount of payload data is sent.  
 \*RST: GO

**Example:**

BB:BTO:PCON:PTYP DH1  
 sets the packet type.  
 BB:BTO:PCON:DSFP PED  
 enable packet editor under data source for packet  
 BB:BTO:PCON:PFC GO  
 allows the flow per logical link.

**Manual operation:** See "[Flow Control](#)" on page 46

**[:SOURCE<hw>]:BB:BTOoth:PCONfiguration:PLENgtH <PLength>**

Sets the packet length in symbols.

**Parameters:**

<PLength> integer  
 Range: 1 to depends on packet type  
 Increment: 1  
 \*RST: depends on packet type

**Example:** BB:BTO:PCON:DSFP ADAT  
fills the all data under data source for packet.  
BB:BTO:PCON:PLEN 1  
sets the packet length.

**Manual operation:** See "[Packet Length](#)" on page 49

**[:SOURCE<hw>]:BB:BT0oth:PCONfiguration:SLAP <State>**

Activates synchronization of the lower address part (LAP) of the sync word and Bluetooth device address.

**Parameters:**  
<State> 1 | ON | 0 | OFF  
\*RST: 1

**Example:** BB:BTO:PCON:SLAP 0  
deactivates LAP synchronization.  
BB:BTO:PCON:LFSW #H000080,24  
sets LAP of the sync word separately.

**Manual operation:** See "[Synchronize LAP with BD\\_ADDR](#)" on page 43

**[:SOURCE<hw>]:BB:BT0oth:PCONfiguration:SNsValue <SnSvalue>**

Sets the start value of the header SEQN bit. The SEQN bit is present in the header to filter out retransmissions in the destination. The signal generator is altering this bit automatically on consecutive frames, if a sequence length of at least 2 frames is set.

**Parameters:**  
<SnSvalue> integer  
Range: 0 to 1  
\*RST: 1

**Example:** BB:BTO:PCON:PTYP DH1  
sets the packet type.  
BB:BTO:PCON:DSFP PED  
enables packet editor under data source for packet.  
BB:BTO:PCON:SNsv ONE  
sets the SEQN bit of the first CRC data packet at the start of a connection.

**Manual operation:** See "[SEQN Start Value](#)" on page 45

**[:SOURCE<hw>]:BB:BT0oth:PCONfiguration:SRMode <SrMode>**

The command indicates the interval between two consecutive page scan windows, determines the behavior of the paging device.

**Parameters:**  
<SrMode> R0 | R1 | R2



**R0**

The scan interval is equal to the scan window T w page scan (continuous nscan) and maximal 1.28s.

**R1**

The scan interval is maximal 1.28s.

**R2**

The scan interval is maximal 2.56s.

\*RST: R0

**Example:**

BB:BTO:PCON:PTYP FHS

sets the packet type.

BB:BTO:PCON:DSFP PED

enables packet editor under data source for packet.

BB:BTO:PCON:SRM R0

sets the scan repetition mode.

**Manual operation:** See "[Scan Repetition Mode](#)" on page 46

**[:SOURCE<hw>]:BB:BTObth:PCONfiguration:VDATA <VData>**

Selects the data source for the voice field.

**Parameters:**

<VData> ALL0 | ALL1 | PATtern | PN09 | PN11 | PN15 | PN16 | PN20 |  
PN21 | PN23 | DLISt

\*RST: PN09

**Example:**

BB:BTO:PCON:VDAT ALL1

sets the voice data type.

**Manual operation:** See "[Data Source \(Voice Field\)](#)" on page 47

## 5.5 Channel configuration commands - LE

**Example: To configure general channel settings for LE**

```
// *****
// Set time unit to ms.
// *****
SOURCE1:BB:BTObth:UNIT:TIME MS

// *****
// Select channel type, test packet, packet format, sequence length,
// controller role. Enable CRC corruption for every second packet.
// *****
SOURCE1:BB:BTObth:CTYPe DATA
SOURCE1:BB:BTObth:UPTYPe TPAC
SOURCE1:BB:BTObth:PFORMat L2M
SOURCE1:BB:BTObth:USLength 2
```

```

SOURce1:BB:BT0oth:BCRole CENTral
SOURce1:BB:BT0oth:CCRC:STATe 1

// *****
// Alternatively set duty cycle for ADV_DIRECT_IND
// *****
SOURce1:BB:BT0oth:CTYPe ADV
SOURce1:BB:BT0oth:UPTYPe ADIN
SOURce1:BB:BT0oth:DCYClE LOW

[:SOURce<hw>]:BB:BT0oth:BCRole.....130
[:SOURce<hw>]:BB:BT0oth:CCRC:STATe..... 130
[:SOURce<hw>]:BB:BT0oth:CTYPe..... 131
[:SOURce<hw>]:BB:BT0oth:DCYClE..... 131
[:SOURce<hw>]:BB:BT0oth:PFORmat..... 131
[:SOURce<hw>]:BB:BT0oth:UPTYPe..... 132
[:SOURce<hw>]:BB:BT0oth:USLength..... 133
[:SOURce<hw>]:BB:BT0oth:MFORmat..... 133
[:SOURce<hw>]:BB:BT0oth:DURation..... 134

```

---

**[:SOURce<hw>]:BB:BT0oth:BCRole <BcRole>**

Determines the controller role.

Depending on the selected channel type different roles are assigned to the controller. For channel type "Data", Central or Peripheral can be assigned. If channel type "Advertising" is selected, the parameter is read only and displayed directly above the graph.

**Parameters:**

<BcRole>                   CENTral | PERipheral | ADvertiser | SCANner | INITiator

**CENTral**

Selects Central as controller role.

**PERipheral**

Selects Peripheral as controller role.

**ADvertiser|SCANner|INITiator**

Assigned roles depending on the selected packet type of the respective channel type.

\*RST:           CENTral

**Example:**                See [Example "To configure general channel settings for LE"](#) on page 129.

**Manual operation:**    See ["Bluetooth Controller Role"](#) on page 53

---

**[:SOURce<hw>]:BB:BT0oth:CCRC:STATe <State>**

Enables/disables the corruption of CRC for every second generated packet. If enabled, only 50% of packets are generated with correct CRC.

**Parameters:**

<State>                    1 | ON | 0 | OFF

\*RST:            0

**Example:** See [Example "To configure general channel settings for LE"](#) on page 129.

**Manual operation:** See ["Corrupted CRC Every 2nd Packet"](#) on page 53

**[:SOURce<hw>]:BB:BT0oth:CTYPe <CType>**

Determines the channel type. Advertising and data are available.

**Parameters:**

<CType>                   ADVertising | DATA

**ADVertising**

Selects channel type advertising.

**DATA**

Selects channel type data. Devices in a connected state transmit data channel packets in connection events with a start point and an interval.

\*RST:           ADVertising

**Example:** See [Example "To configure general channel settings for LE"](#) on page 129.

**Manual operation:** See ["Channel Type"](#) on page 51

**[:SOURce<hw>]:BB:BT0oth:DCYClE <DCycle>**

Specifies duty cycle for directed advertising (packet type ADV\_DIRECT\_IND).

**Parameters:**

<DCycle>                   LOW | HIGH

\*RST:           HIGH

**Example:** See [Example "To configure general channel settings for LE"](#) on page 129.

**Options:**               R&S SMM-K117

**Manual operation:** See ["Duty Cycle"](#) on page 52

**[:SOURce<hw>]:BB:BT0oth:PFORmat <PFormat>**

Specifies the physical layer of LE signal.

**Parameters:**

<PFormat>               L1M | L2M | LCOD

**L1M**

LE 1M

**L2M**

LE 2M

**LCOD**

LE coded

\*RST: L1M

**Example:** See [Example "To configure general channel settings for LE"](#) on page 129.

**Options:** R&S SMM-K117 required for L2M, LCOD

**Manual operation:** See ["Packet Format"](#) on page 52

**[<SOURce<hw>]:BB:BT0oth:UPType <UpType>**

Selects the packet type. The available packets depend on the selected channel type and installed options.

The tables below provide an overview. For more information, see data sheet.

*Table 5-1: R&S SMM-K60 packet/PDU types*

<UpType>	Packet/PDU type	<UpType>	Packet/PDU type
AIND	ADV_IND	ERSP	LL_ENC_RSP
ADINd	ADV_DIRECT_IND	SEReq	LL_START_ENC_REQ
ANINd	ADV_NONCONN_IND	SERSp	LL_START_ENC_RSP
SREQ	SCAN_REQ	URSP	LL_UNKNONW_RSP
SRSP	SCAN_RSP	FREQ	LL_FEATURE_REQ
CREQ	CONNECT_IND	FRSP	LL_FEATURE_RSP
ADCind	ADV_SCAN_IND	TPACket	TEST PACKET
DATA	DATA	PEReq	LL_PAUSE_ENC_REQ
CUReq	LL_CONNECTION_UPDATE_IND	PERSp	LL_PAUSE_ENC_RSP
CMReq	LL_CHANNEL_MAP_IND	VIND	LL_VERSION_IND
TIND	LL_TERMINATE_IND	RIND	LL_REJECT_IND
EREQ	LL_ENC_REQ		

*Table 5-2: R&S SMM-K117 packet/PDU types*

<UpType>	Packet/PDU type	<UpType>	Packet/PDU type
PREQ	LL_PHY_REQ	ASINd	AUX_SYNC_IND
PRSP	LL_PHY_RSP	ASReq	AUX_SCAN_REQ
PUIN	LL_PHY_UPDATE_IND	ASPSp	AUX_SCAN_RSP
LRSP	LL_LENGTH_RSP	ACRSp	AUX_CONNECT_RSP
SFR	LL_PERIPHERAL_FEATURE_REQ	ACReq	AUX_CONNECT_REQ
CPR	LL_CONNECTION_PARAM_REQ	MUCH	LL_MIN_USED_CHANNELS_IND
CPRS	LL_CONNECTION_PARAM_RSP	CONT	CONTINUOUS
REIN	LL_REJECT_EXT_IND	CTEQ	LL_CTE_REQ
PIR	LL_PING_REQ	CTEP	LL_CTE_RSP
PIRS	LL_PING_RSP	PSIND	LL_PERIODIC_SYNC
AEINd	ADV_EXT_IND	CAReq	LL_CLOCK_ACCURACY_REQ
AAINd	AUX_ADV_IND	CARSp	LL_CLOCK_ACCURACY_RSP
ACINd	AUX_CHAIN_IND		

**Parameters:**

<UpType> AIND | ADINd | ANINd | SREQ | SRSP | CREQ | ADCind |  
 DATA | CUReq | CMReq | TIND | EREQ | ERSP | SEReq |  
 SERSp | URSP | FREQ | FRSP | TPACket | PEReq | PERSp |  
 VIND | RIND | PREQ | PRSP | PUIN | LREQ | LRSP | SFR |  
 CPR | CPRS | REIN | PIR | PIRS | AEINd | AAInD | ACINd |  
 ASINd | ASReq | ASPSp | ACRSp | ACReq | MUCH | CONT |  
 CTEQ | CTEP | PSINd | CAReq | CARSp  
 \*RST: AIND

**Example:**

See [Example "To configure general channel settings for LE"](#) on page 129.

**Manual operation:** See ["Packet Type"](#) on page 51

**[:SOURce<hw>]:BB:BT0oth:USLength <UsLength>**

Selects the number of frames or events depending on the packet type. The signal repeats after the specified number of frames/events.

For SCAN\_REQ and CONNECT\_IND packet, the sequence length is expressed in "Frames".

For AUX\_SCAN\_REQ and AUX\_CONNECT\_REQ packet, the sequence length is expressed in "Frames".

For LL\_TERMINATE\_IND packets, a default value according to the specification is given:

Central: PeripheralLatency + 6

Peripheral: 6

For all other packet types the sequence length is expressed in "Events".

**Parameters:**

<UsLength> integer  
 Range: depends on the number of states in dirty transmitter  
 test to dynamic  
 \*RST: 1

**Example:**

See [Example "To configure general channel settings for LE"](#) on page 129.

**Manual operation:** See ["Sequence Length"](#) on page 53

**[:SOURce<hw>]:BB:BT0oth:MFORmat <ModFmt>**

Specifies the physical layer used for CONTINUOUS payload transmission.

**Parameters:**

<ModFmt> L1M | L2M | LCOD  
**L1M:** LE 1M  
**L2M:** LE 2M  
**LCOD:** LE coded  
 \*RST: L1M

**Example:** See [Example "To configure general channel settings for LE"](#) on page 129.

**Options:** R&S SMM-K117 required for L2M, LCOD

**Manual operation:** See ["Modulation Format"](#) on page 54

**[:SOURce<hw>]:BB:BT0oth:DURation <Duration>**

Specifies the transmission duration of CONTINUOUS payload transmission.

Command sets the values in ms. Query returns values in s.

**Parameters:**

<Duration> float  
 Range: depending on modulation format, symbols per a bit and payload type  
 Default unit: ms

**Example:** See [Example "To configure general channel settings for LE"](#) on page 129.

**Manual operation:** See ["Duration"](#) on page 54

## 5.6 Event and frame configuration commands - LE

### Example: Configure event and frame configuration settings

```
// *****
// For ADV_SCAN_IND, select advertising event interval.
// *****
SOURcel:BB:BT0oth:CTYPe ADV
SOURcel:BB:BT0oth:UPTYPe ADC
SOURcel:BB:BT0oth:ECONfiguration:AEINterval 20

// *****
// Alternatively set advertising event interval for ADV_DIRECT_IND.
// *****
SOURcel:BB:BT0oth:CTYPe ADV
SOURcel:BB:BT0oth:UPTYPe ADIN
SOURcel:BB:BT0oth:ECONfiguration:ADINterval 3.75
// *****
// Set advertising event delay, activate channel 37.
```

## Event and frame configuration commands - LE

```

// *****
SOURCEl:BB:BT0oth:ECOnfiguration:AEDElay 0
SOURCEl:BB:BT0oth:ECOnfiguration:ACTable:CHANnel0:STATE 1
// *****
// For packet type SCAN_REQ, set length of the window,
// time interval.
// *****
SOURCEl:BB:BT0oth:CTYPe ADV
SOURCEl:BB:BT0oth:UPTYpe SREQ
SOURCEl:BB:BT0oth:ECOnfiguration:SWINdow 10
SOURCEl:BB:BT0oth:ECOnfiguration:SINTErval 3500

// *****
// For packet type CONNECT_IND set transmit window,
// start point of the transmit window,
// LL connection timeout and time interval.
// *****
SOURCEl:BB:BT0oth:CTYPe ADV
SOURCEl:BB:BT0oth:UPTYpe CREQ
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:WSIZE 8.25
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:WOffset 800
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:LCtimeout 7500
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:CINTErval 6400

// *****
// Alternatively select packet type LL_CONNECTION_UPDATE_IND,
// set transmit window, start point of the transmit
// window, LL connection timeout and time interval.
// *****
SOURCEl:BB:BT0oth:CTYPe DATA
SOURCEl:BB:BT0oth:UPTYpe CUR
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:NWSize 8.25
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:NWOffset 800
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:NLCTimeout 7500
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:NCINTErval 6400

// *****
// Set time interval for advertising channel and events.
// Set time delay for advertising events.
// *****
SOURCEl:BB:BT0oth:ECOnfiguration:APInterval 1.3
SOURCEl:BB:BT0oth:ECOnfiguration:AEINTErval 15
SOURCEl:BB:BT0oth:ECOnfiguration:AEDElay 5

```

[ :SOURCE<hw> ]:BB:BT0oth:ECOnfiguration:ACTable:CHANnel<ch0>:STATE.....	136
[ :SOURCE<hw> ]:BB:BT0oth:ECOnfiguration:DCTable:CHANnel<ch0>:STATE.....	136
[ :SOURCE<hw> ]:BB:BT0oth:ECOnfiguration:PCOnfiguration:DCMTable: CHANnel<ch0>:STATE.....	136
[ :SOURCE<hw> ]:BB:BT0oth:ECOnfiguration:AEDElay.....	137
[ :SOURCE<hw> ]:BB:BT0oth:ECOnfiguration:ADINTErval.....	137
[ :SOURCE<hw> ]:BB:BT0oth:ECOnfiguration:AEINTErval.....	137
[ :SOURCE<hw> ]:BB:BT0oth:ECOnfiguration:APInterval.....	138

<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:LCMode</code> .....	138
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:LTKey</code> .....	139
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:MNINterval</code> .....	139
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:MXINterval</code> .....	140
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:OFFSet&lt;ch0&gt;:STATe</code> .....	140
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:OFFSet&lt;ch0&gt;:VALue</code> .....	140
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:PAINterval</code> .....	141
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:PPERiodicity</code> .....	141
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:RCECount</code> .....	142
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PNUMber</code> .....	142
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:SDCI?</code> .....	142
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:SINterval</code> .....	143
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:SWINdow</code> .....	143
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:WOINfo?</code> .....	143
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:WSINfo?</code> .....	144

---

`[:SOURce<hw>]:BB:BT0oth:ECONfiguration:ACTable:CHANnel<ch0>:STATe`  
<State>

`[:SOURce<hw>]:BB:BT0oth:ECONfiguration:DCTable:CHANnel<ch0>:STATe`  
<State>

`[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:DCMTTable:`  
`CHANnel<ch0>:STATe` <State>

Indicates used and unused data channels.

**Note:** The previously used syntax `...:SET<ch>:STATe` has been replaced by `...:CHANnel<ch>:STATe`. Compatibility to the previous commands is given.

This parameter is relevant for data event and advertising frame configuration with the packet types `LL_CHANNEL_MAP_IND`, `CONNECT_IND`.

Within the option R&S SMM-K117, the following packet types are also relevant for the setting: `AUX_CONNECT_IND`, `AUX_EXT_IND`, `AUX_ADV_IND`, `AUX_CHAIN_IND`, `AUX_SYNC_IND`, `AUX_SCAN_RSP`.

**Parameters:**

<State>                    1 | ON | 0 | OFF  
\*RST:                    0

**Example:**

```
SOURce1:BB:BT0:ECON:ACT:CHAN:STAT ON
State in Advertising Channel Table and Secondary Advertising
Channel Table
SOURce1:BB:BT0:ECON:DCT:CHAN:STAT ON
State in Data Channel Table and Secondary Advertising Chan-
nel Table
SOURce1:BB:BT0:ECON:PCON:DCMT:CHAN:STAT ON
State in Data Channel Map Table
```

**Example:** See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Manual operation:** See ["Channel Table"](#) on page 61



---

**[[:SOURce<hw>]:BB:BT0oth:ECONfiguration:AEDElay <AeDelay>**

Sets a time delay between the start times of two consecutive advertising events. The value is added to the advertising event interval.

Command sets the values in ms. Query returns values in s.

**Parameters:**

<AeDelay> float  
 Range: 0 s to 10E-3 s  
 Increment: 0.1E-3  
 \*RST: 0  
 Default unit: ms

**Example:** SOURce1:BB:BT0:ECON:AEDElay 5  
 Sets a delay of 5 ms.

**Example:** See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Manual operation:** See ["Advertising Event Delay"](#) on page 56

---

**[[:SOURce<hw>]:BB:BT0oth:ECONfiguration:ADINterval <AdInterval>**

Sets the time interval between two consecutive advertising events for packet type "ADV\_DIRECT\_IND" and duty cycle high.

Command sets the values in ms. Query returns values in s.

**Parameters:**

<AdInterval> float  
 Range: 1.05E-3 s to 3.75E-3 s  
 Increment: 0.01E-3  
 \*RST: 3.75E-3  
 Default unit: ms

**Example:** SOURce1:BB:BT0:ECON:ADIN 13  
 Sets a time interval of 13 ms.

**Example:** See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Manual operation:** See ["Advertising Event Interval"](#) on page 56

---

**[[:SOURce<hw>]:BB:BT0oth:ECONfiguration:AEINterval <AeInterval>**

Sets the time interval between two consecutive advertising events, with regard to the starting points.

Command sets the values in ms. Query returns values in s.

**Parameters:**

<AeInterval> float  
 Range: 5E-3 s to depends on oversampling  
 Increment: 0.1E-3  
 \*RST: 20E-3  
 Default unit: ms

**Example:**

SOURCE1:BB:BTO:ECON:AEIN 15  
 Sets a time interval of 15 ms.

**Example:**

See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Manual operation:**

See ["Advertising Event Interval"](#) on page 56

**[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:APInterval <ApInterval>**

Sets the time interval between packets starting points of two consecutive packets in the advertising channel.

**Parameters:**

<ApInterval> float  
 Range: 1.3E-3 to 28E-3  
 Increment: 0.1E-3  
 \*RST: 10E-3  
 Default unit: ms

**Example:**

SOURCE1:BB:BTO:ECON:APIN 1.3  
 Sets a time interval of 1.3 ms.

**Example:**

See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Manual operation:**

See ["Advertising Packet Interval"](#) on page 57

**[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:LCMode <LcMode>**

Selects the link layer connection mode. In order to provide safe transmission of payload data, the data in the packet can be encrypted. If activated, the payload data follows MIC (Message authentication Code).

**Parameters:**

<LcMode> UENC | ENC  
**UENC**  
 Payload data is transmitted without encoding.  
**ENC**  
 The link layer connection runs in encrypted mode.  
 \*RST: UENC

**Example:**            `SOURce1:BB:BTO:ECON:LCM UENC`  
without encoding.  
`SOURce1:BB:BTO:ECON:LCM ENC`  
in encrypted mode.

**Example:**            See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Manual operation:** See "[LL Connection Mode](#)" on page 59

**[:SOURce<hw>]:BB:BTOoth:ECONfiguration:LTKey <LtKey>, <BitCount>**

Indicates the time the controller needs to receive the long-term key from the host. After this time, the controller is ready to enter into the last phase of encryption mode setup.

**Parameters:**

<LtKey>                numeric  
                          \*RST:        #H0

<BitCount>            integer  
                          Range:        128 to 128  
                          \*RST:        128

**Example:**            `SOURce1:BB:BTO:ECON:LCM ENC`  
`SOURce1:BB:BTO:ECON:LTK`  
`#H00000000000000000000000000000000,128`  
In encrypted mode, the code can be edited.

**Example:**            See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Manual operation:** See "[Long Term key \(hex\)](#)" on page 60

**[:SOURce<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:MNInterval <MNInterval>**

Specifies the minimum allowed connection interval.

Command sets the values in ms. Query returns values in s.

**Parameters:**

<MNInterval>         float  
                          Range:        7.5E-3 s to depending on Max. Interval  
                          Increment:   1.25E-3 s  
                          \*RST:        7.5E-3

**Example:**            `SOURce1:BB:BTO:ECON:PCON:MNIN 7.5`  
Sets a time interval of 7.5 ms.

**Example:**            See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Options:**             R&S SMM-K117

**Manual operation:** See "[Min. / Max. Interval](#)" on page 78

---

**[[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MXINterval  
<MInterval>**

Specifies the maximum allowed connection interval.

Command sets the values in ms. Query returns values in s.

**Parameters:**

<MInterval> float  
 Range: 7.5E-3 s to 4000E-3 s  
 Increment: 1.25E-3  
 \*RST: 7.5E-3

**Example:** SOURce1:BB:BT0:ECON:PCON:MXIN 12.5  
 Sets a time interval of 12.5 ms.

**Example:** See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Options:** R&S SMM-K117

**Manual operation:** See "[Min. / Max. Interval](#)" on page 78

---

**[[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:OFFSet<ch0>:  
STATE <State>**

Enables / disables Offset0 to Offset5 of the offset setting table.

**Parameters:**

<State> 1 | ON | 0 | OFF  
 \*RST: 0

**Example:** See [Example"Configure event and frame configuration settings"](#) on page 134.

**Options:** R&S SMM-K117

**Manual operation:** See "[Offset Setting Table](#)" on page 79

---

**[[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:OFFSet<ch0>:  
VALue <Offset>**

Specifies Offset0 to Offset5 of the offset setting table.

Command sets the values in ms. Query returns values in s.

**Parameters:**

<Offset> float  
 Range: 0 s to depending on Max. Interval  
 Increment: 1.25  
 \*RST: 0  
 Default unit: ms

**Example:** `SOURce1:BB:BT0:ECON:PCON:OFFSet0:VALue 7`  
`SOURce1:BB:BT0:ECON:PCON:OFFSet0:VALue?`  
`// Response: "7.5"`  
 Sets the Offset0 to 7.5 ms. The setting 7 ms is automatically changed to the closest multiple of 1.25 ms, which is 7.5 ms.

**Example:** See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Options:** R&S SMM-K117

**Manual operation:** See "[Offset Setting Table](#)" on page 79

**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PAInTerval**  
 <Interval>

Sets the time interval between the start of two AUX\_SYNC\_IND PDUs from the same advertising set.

Command sets the values in ms. Query returns values in s.

**Parameters:**

<Interval> float  
 Range: 7.5E-3 s to depending on oversampling  
 Increment: 0.01E-3  
 \*RST: 20E-3  
 Default unit: ms

**Example:** `SOURce1:BB:BT0oth:ECON:PCON:PAInTerval 10`  
 Sets a time interval of 10 ms.

**Example:** See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Manual operation:** See "[Periodic Advertising Interval](#)" on page 56

**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PPERiodicity**  
 <PPERiodicity>

Specifies a value the connection interval is preferred to be a multiple of.

**Parameters:**

<PPERiodicity> float  
 Range: 0 to depends on Max. Interval  
 Increment: 0.01E-3  
 \*RST: 0

**Example:** See [Example"Configure event and frame configuration settings"](#) on page 134.

**Options:** R&S SMM-K117

**Manual operation:** See "[Preferred Periodicity](#)" on page 78

---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:RCECount**  
<RCECount>

Specifies the ReferenceConnEventCount field of LL\_CONNECTION\_PARAM\_REQ.

**Parameters:**

<RCECount>            integer  
                          Range:     0 to 65535  
                          \*RST:     0

**Example:**            See [Example"Configure event and frame configuration settings"](#) on page 134.

**Options:**            R&S SMM-K117

**Manual operation:** See ["Ref. Connection Event Count"](#) on page 78

---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PNUMber** <PNumber>

Sets the number of Tx packets per event.

Each connection contains at least one data channel packet. The maximum number of packets per event is determined by the duration of the connection event interval.

**Parameters:**

<PNumber>            integer  
                          Range:     1 to depends on connection event interval  
                          \*RST:     1

**Example:**            SOURce1:BB:BT0:ECOn:PNUM 2580  
                          Sets the number of Tx packets per event.

**Example:**            See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Manual operation:** See ["No. Of Tx Packets/Event"](#) on page 58

---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:SDCI?**

Queries the number of the first active data channel.

**Return values:**

<SelectedChannel>   integer  
                          Range:     0 to 36  
                          \*RST:     0

**Example:**            SOURce1:BB:BT0oth:ECOnfiguration:SDCI?

**Example:**            See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Usage:**            Query only

**Manual operation:** See ["Selected Data Channel Index"](#) on page 60

**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:SINTerval <SInterval>**

Sets the time interval between the starting points of two consecutive windows during which the scanner is operating in an advertising channel.

Command sets the values in ms. Query returns values in s.

**Parameters:**

<SInterval> float  
 Range: 10E-3 s to depends on oversampling and the number of advertising channel table states  
 Increment: 0.625E-3  
 \*RST: 10E-3  
 Default unit: ms

**Example:** `SOURce1:BB:BT0:ECON:SINT 3.5`  
 Sets a time interval of 3.5 ms.

**Example:** See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Manual operation:** See ["Scan Interval"](#) on page 57

**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:SWINdow <SWindow>**

Sets the length of the window during which the scanner is operating in the advertising channel.

Note that the scan window is less or equal to the value of the scan interval.

Command sets the values in ms. Query returns values in s.

**Parameters:**

<SWindow> float  
 Range: 10E-3 s to 10240E-3 s  
 Increment: 0.625E-3  
 \*RST: 10E-3  
 Default unit: ms

**Example:** `SOURce1:BB:BT0:ECON:SWIN 10`  
 Sets the length of the window to 10 ms.

**Example:** See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Manual operation:** See ["Scan Window"](#) on page 57

**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:WOINfo?**

Requires data event and advertising frame configuration with the packet type CONNECT\_IND.

Queries the start point of the transmit window.

**Return values:**

<WolInfo> string

**Example:**

```
SOURce1:BB:BTO:UPTY CREQ
Sets packet type CONNECT_IND
SOURce1:BB:BTO:ECON:PCON:WOIN?
Queries the start point of the transmit window.
```

**Example:**

See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Usage:**

Query only

**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:WSINfo?**

Requires data event and advertising frame configuration with the packet type CONNECT\_IND.

Queries the size of the transmit window, regarding to the start point.

**Return values:**

<WsInfo> string

**Example:**

```
SOURce1:BB:BTO:UPTY CREQ
Sets packet type CONNECT_IND
SOURce1:BB:BTO:ECON:PCON:WSIN?
Queries the size of the transmit window.
```

**Example:**

See also [Example"Configure event and frame configuration settings"](#) on page 134.

**Usage:**

Query only

**Manual operation:** See ["Transmit Window Size"](#) on page 58

## 5.7 Packet configuration commands - LE

**Example: To configure advertising packets**

```
// *****
// Configure packet for ADV_SCAN_IND: switch off whitening,
// set Tx device address type, AdvA, data source and length
// *****
SOURce1:BB:BT0oth:CTYPe ADV
SOURce1:BB:BT0oth:UPTYPe ADC
SOURce1:BB:BT0oth:ECONfiguration:PCONfiguration:DWhitening 0
SOURce1:BB:BT0oth:ECONfiguration:PCONfiguration:TATYPe PUBL
SOURce1:BB:BT0oth:ECONfiguration:PCONfiguration:ANUap #H017412,24
SOURce1:BB:BT0oth:ECONfiguration:PCONfiguration:ALAP #H9E8B00,24
SOURce1:BB:BT0oth:ECONfiguration:PCONfiguration:DATA PN09
SOURce1:BB:BT0oth:ECONfiguration:PCONfiguration:DLENGth 31
```



## Packet configuration commands - LE

```

// *****
// Alternatively set periodic advertising interval for AUX_SYNC_IND
// *****
SOURCEl:BB:BT0oth:CTYPe ADV
SOURCEl:BB:BT0oth:UPTYpe ASIN
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:PAINterval

// *****
// Select packet type AUX_ADV_IND. (All parameters of
// extended header are configurable with AUX_ADV_IND packet
// type.) Set channel selection,
// Tx device address type, advertising mode.
// Enable extended header for advertising PDUs. Enable all
// flags: AdvA, TargetA, AdvData Info, Aux Ptr, Sync Info,
// TxPow. Set Tx power value and ACAD length and pattern.
// *****
SOURCEl:BB:BT0oth:CTYPe ADV
SOURCEl:BB:BT0oth:UPTYpe AAIN
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:CSElection CS1
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:TATYPe PUBL
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:AMODE NCNS
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:EHEader:STATE ON
SOURCEl:BB:BT0oth:ECONfiguration:PCON:EHFLags:AADDRESS:STATE ON
SOURCEl:BB:BT0oth:ECONfiguration:PCON:EHFLags:TADDRESS:STATE ON
SOURCEl:BB:BT0oth:ECONfiguration:PCON:EHFLags:ADINfo:STATE ON
SOURCEl:BB:BT0oth:ECONfiguration:PCON:EHFLags:APTR:STATE ON
SOURCEl:BB:BT0oth:ECONfiguration:PCON:EHFLags:SINfo:STATE ON
SOURCEl:BB:BT0oth:ECONfiguration:PCON:EHFLags:TPOWER:STATE ON
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:TPOWER -100
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:ALENgtH 16
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:ACAD PN16

// *****
// Alternatively set user-defined pattern.
// *****
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:ACAD PATTern
SOURCEl:BB:BT0oth:ECONfiguration:PCON:ACAD:APATtern #B011000011,9
// *****
// Alternatively set pattern from data list.
// *****
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:ACAD DLISt
SOURCEl:BB:BT0oth:ECON:PCON:ACAD:ASEL "p:/pattern1.dm_iqd

// *****
// Enable AdvData Info flag, set advertising data ID and
// advertising data set ID.
// *****
SOURCEl:BB:BT0oth:ECONfiguration:PCON:EHFLags:ADINfo:STATE ON

```

## Packet configuration commands - LE

```

SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:ADID #H01FF,12
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:ASID #H0F,4
// *****
// Enable and specify AuxPtr for the secondary advertising
// channel: select AUX channel, set clock accuracy, offset unit,
// AUX offset, AUX PHY.
// *****
SOURCEl:BB:BT0oth:ECOnfiguration:PCOn:EHFLags:APTR:STATe ON
SOURCEl:BB:BT0oth:ECOnfiguration:DCTable:CHANnel0:STATe 1
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:CACCuracy T500
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:AOUNits U30
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:AOffset 100
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:APHY L1M
// *****
// Enable and specify SyncInfo field for the AUX_SYNC_IND
// packet: set sync packet offset, offset unit, periodic adv
// interval, sleep clock accuracy, access address, CRC initial
// value. Reset event counter. Configure channel map.
// *****
SOURCEl:BB:BT0oth:ECOnfiguration:PCOn:EHFLags:SIInfo:STATe ON
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:PAInTerval 20
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:SCACCuracy SCA0
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:AADdress #HACDE48AC,32
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:CIV #H000000,24
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:SPOffset 100
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:SOUNits U30
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:OADJust 1
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:SPOffset?
// 99.9
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:ECOUNTER 0
SOURCEl:BB:BT0oth:ECOnfiguration:PCOn:DCMTable:CHANnel0:STATe 1
// *****
// Configure advertiser, scanner and initiator device addresses.
// Company_Assigned and Company_Id in Advertiser's Device Address
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:ACAS #H000000,24
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:ACID #H000000,24
// Company_Assigned and Company_Id in Scanners Device Address
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:SCAS #H000000,24
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:SCID #H000000,24
// Company_Assigned and Company_Id in Initiators Device Address
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:ICAS #H000000,24
SOURCEl:BB:BT0oth:ECOnfiguration:PCOnfiguration:ICID #H000000,24
// *****

```

**Example: To configure data packets**

```
// *****
// Configure signal for DATA: select channel type, packet type,
// packet format, sequence length, role, enable CRC corruption.
// *****
SOURCEl:BB:BT0oth:CTYPe DATA
SOURCEl:BB:BT0oth:UPTYPe CUR
SOURCEl:BB:BT0oth:PFORmat L1M
SOURCEl:BB:BT0oth:USLength 12
SOURCEl:BB:BT0oth:BCRole CENTRAL
SOURCEl:BB:BT0oth:CCRC:STATe 1

// *****
// Configure packet type DATA: access address, CRC initial,
// NESN start, SN start values, data source, data length
// *****
SOURCEl:BB:BT0oth:CTYPe DATA
SOURCEl:BB:BT0oth:UPTYPe DATA
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:AADDRESS #HDAB85479,32
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:CIVAlue #H000000,24
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:NSValue 0
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:SSValue 0
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:DATA PN09
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:DLENGth 251

// *****
// Configure payload of LL_CONNECTION_UPDATE_IND: transmit
// window size and offset, connection event interval, Peripheral
// latency, LL connection timeout, connection instant.
// *****
SOURCEl:BB:BT0oth:CTYPe DATA
SOURCEl:BB:BT0oth:UPTYPe CUR
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:AADDRESS #HDAB85479,32
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:CIVAlue #H000000,24
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:NSValue 0
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:SSValue 0
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:NWSize 1.25
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:NWOffset 0
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:NCINterval 7.5
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:PSLatency 0
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:NLCTimeout 100
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:CINStant 0

// *****
// Configure the payload of LL_FEATURE_REQ: set packet
// type, feature set length and configure feature set.
// *****
SOURCEl:BB:BT0oth:CTYPe DATA
SOURCEl:BB:BT0oth:UPTYPe FREQ
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:FSLength 8
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:FSBit0:STATe 1
```

## Packet configuration commands - LE

```

SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit1:STATe 1
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit2:STATe 1
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit3:STATe 1
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit4:STATe 1
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit5:STATe 1
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit6:STATe 1
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit7:STATe 0
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit8:STATe 1
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit9:STATe 0
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit10:STATe 0
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit11:STATe 0
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit12:STATe 0
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit13:STATe 0
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit14:STATe 1
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit15:STATe 0
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:FSBit16:STATe 0

// *****
// Configure the payload of LL_CONNECTION_PARAM_REQ: max. and min.
// interval, Peripheral latency, LL connection timeout, preferred
// periodicity, reference connection even count
// *****
SOURCEl:BB:BT00th:CTYPe DATA
SOURCEl:BB:BT00th:UPTYpe CPR
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:MXINterval 10
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:MNINterval 7.5
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:PSLatency 1
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:NLCTimeout 100
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:PPERiodicity 1.25
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:RCECount 10
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:OFFSet0:STATe ON
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:OFFSet0:VALue 3.75
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:OFFSet1:STATe ON
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:OFFSet1:VALue 5
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:OFFSet2:STATe ON
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:OFFSet2:VALue 6.25
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:OFFSet3:STATe OFF
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:OFFSet4:STATe OFF
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:OFFSet5:STATe OFF

// *****
// Configure the payload of LL_LENGTH_REQ: max. Rx and TX
// payload octets, max. time to receive and transmit a packet.
// *****
SOURCEl:BB:BT00th:CTYPe DATA
SOURCEl:BB:BT00th:UPTYpe LREQ
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:MROctets 27
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:MTOctets 27
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:MRTime 0.328
SOURCEl:BB:BT00th:ECONfiguration:PCONfiguration:MTTime 0.328

```

## Packet configuration commands - LE

```

// *****
// Set the payload of LL_PHY_REQ: specify preferred Tx, Rx PHYs.
// *****
SOURCEl:BB:BT0oth:CTYPe DATA
SOURCEl:BB:BT0oth:UPTYPe PREQ
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:TPHYs:L1M:STATe 1
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:TPHYs:L2M:STATe 1
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:TPHYs:LCOD:STATe 0
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:RPHYs:L1M:STATe 1
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:RPHYs:L2M:STATe 1
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:RPHYs:LCOD:STATe 0

// *****
// Set the payload of LL_PHY_UPDATE_IND: specify PHYs for
// Central-to-Peripheral and Peripheral-to-Central direction and instant.
// *****
SOURCEl:BB:BT0oth:CTYPe DATA
SOURCEl:BB:BT0oth:UPTYPe PUIN
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:MTSPHy:L1M:STATe 1
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:MTSPHy:L2M:STATe 1
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:MTSPHy:LCOD:STATe 0
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:STMPHy:L1M:STATe 1
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:STMPHy:L2M:STATe 1
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:STMPHy:LCOD:STATe 0
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:CINStant 1

// *****
// Set the payload of LL_REJECT_EXT_IND: set reject opcode
// *****
SOURCEl:BB:BT0oth:CTYPe DATA
SOURCEl:BB:BT0oth:UPTYPe REIN
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:ROPCode #H02,8

// *****
// Set the payload of LL_MIN_USED_CHANNELS_IND: set the controller
// role to Peripheral, set the packet type, specify PHYs and minimum
// used channels requirement
// *****
SOURCEl:BB:BT0oth:CTYPe DATA
SOURCEl:BB:BT0oth:BCRole PERipheral
SOURCEl:BB:BT0oth:UPTYPe MUCH
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:PHYS:L1M:STATe 1
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:PHYS:L2M:STATe 1
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:PHYS:LCOD:STATe 0
SOURCEl:BB:BT0oth:ECONfiguration:PCONfiguration:MUCHannels 2

// *****
// Set header and payload of LL_PERIODIC_SYNC_IND: enable CTE and

```

```
// configure CTE method, set event counter properties, specify PHYs
// and address type.
// *****
:SOURcel:BB:BT0oth:CTYPe DATA
:SOURcel:BB:BT0oth:UPTYPe PSIN
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:CPResent 1
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:CTIME 0.016
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:CTYPe AOA
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:ID #HAAAA,16
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:SPOffset 245.7
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:SOUNits U30
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:OADJust 1
// Enabling offset adjust sets the Sync packet offset to 300 us.
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:SOUNits U300
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:CECount 65535
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:LPECOUNTER 65535
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:SID #H1,4
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:ATYPe PUBL
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:SCACcuracy SCA0
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:PHY:L1M:STATe 1
// Enabling another PHY automatically disables the previous PHY.
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:PHY:L2M:STATe 1
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:PHY:LCOD:STATe 1
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:PHY:L1M:STATe? 0
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:PHY:L2M:STATe? 0
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:ACASigned #H000080,24
:SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:SCECounter 65535
```

- [General configuration](#)..... 150
- [Header configuration](#)..... 151
- [Payload configuration](#)..... 154

### 5.7.1 General configuration

<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:DWHitening</a> .....	150
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:AADDRESS</a> .....	151
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:CIValue</a> .....	151

---

#### **[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:DWHitening** <DWhitening>

Activates or deactivates the Data Whitening. Evenly distributed white noise is ideal for the transmission and real data can be forced to look similar to white noise with different methods called Data Whitening. Applied to the PDU and CRC fields of all packet types, whitening is used to avoid long equal sequences in the data bit stream.

#### **Parameters:**

<DWhitening>            1 | ON | 0 | OFF  
\*RST:                    0

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Data Whitening"](#) on page 63

**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:AADDRESS**  
 <AAddress>, <BitCount>

Sets the access address of the link layer connection (32-bit string).

**Parameters:**

<AAddress>	numeric
	*RST: #HACDE48AC
<BitCount>	integer
	Range: 32 to 32
	*RST: 32

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Manual operation:** See ["Access Address"](#) on page 63

**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:CIValue** <CiValue>,  
 <BitCount>

Sets the initialization value for the CRC (Cyclic Redundancy Check, 24 bits) calculation. A packet has been received correctly, when it has passed the CRC check.

**Parameters:**

<CiValue>	numeric
	*RST: #H0
<BitCount>	integer
	Range: 24 to 24
	*RST: 24

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Manual operation:** See ["CRC Initial"](#) on page 65

## 5.7.2 Header configuration

<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:CSElection</a> .....	152
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:NSValue</a> .....	152
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:SSValue</a> .....	152
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:CPResent</a> .....	152
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:CTIME</a> .....	153
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:CTReq</a> .....	153
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:CTYPE</a> .....	153
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfig:PCONfig:ANTGain&lt;ch0&gt;</a> .....	154
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfig:PCONfig:ANTNumber</a> .....	154

---

**[ :SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CSElection**  
 <CSelection>

Specifies the algorithm of channel selection.

**Parameters:**

<CSelection> CS1 | CS2  
 Algorithm #1 or algorithm #2  
 \*RST: CS1

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["Channel Selection"](#) on page 64

---

**[ :SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:NSValue** <NsValue>

Sets the start value of the next expected packet from the same device in the LL connection ("N"ext"E"xpected "S"equence"N"umber). This parameter can be set in the first event. From the second event this field is not indicated.

**Parameters:**

<NsValue> integer  
 Range: 0 to 1  
 \*RST: 1

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["NESN Start Value"](#) on page 64

---

**[ :SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:SSValue** <SsValue>

Sets the sequence number of the packet. This parameter can be set in the first event. From the second event, this field is not indicated.

**Parameters:**

<SsValue> integer  
 Range: 0 to 1  
 \*RST: 0

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["SN Start Value"](#) on page 64

---

**[ :SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CPrezent** <State>

Activates the CTEInfo field in the header of Bluetooth LE data packets in the LE un-coded PHY.

**Parameters:**

<State> 1 | ON | 0 | OFF  
 \*RST: 0



**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["CTEInfo Present"](#) on page 65

**[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:CTIME <CTime>**

Sets the CTETime comprising the length of constant tone extension field of the Bluetooth LE PDU.

**Parameters:**

<CTime> float  
 Range: 16E-6 to 160E-6  
 Increment: 8E-6  
 \*RST: 16E-6

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["CTETime"](#) on page 65

**[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:CTReq <CTReq>**

Sets the CTE type in the CTETypeReq field of the CtrData field of the LL\_CTE\_REQ PDU.

**Parameters:**

<CTReq> AOD1 | AOA | AOD2  
**AOA**  
 AoA Constant Tone Extension  
**AOD1**  
 AoD Constant Tone Extension with 1  $\mu$ s time slots  
**AOD2**  
 AoD Constant Tone Extension with 2  $\mu$ s time slots  
 \*RST: AOA

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["CTETypeReq"](#) on page 83

**[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:CTYPE <CType>**

Sets the type of constant tone extension. The type specifies the CTE AoA/AoD method and for AoD the length of the switching and I/Q sampling slots.

**Parameters:**

<CType> AOD1 | AOA | AOD2  
**AOA**  
 AoA Constant Tone Extension  
**AOD1**  
 AoD Constant Tone Extension with 1  $\mu$ s time slots

**AOD2**AoD Constant Tone Extension with 2  $\mu$ s time slots

\*RST: AOA

**Example:** See [Example "To configure data packets"](#) on page 147.**Manual operation:** See ["CTEType"](#) on page 66**[[:SOURce<hw>]:BB:BT0oth:ECONfig:PCONfig:ANTGain<ch0> <AntennaGain>**

Specifies the gain of the antenna. You can specify the antenna gain information of up to four individual antennas for direction finding.

**Parameters:**

<AntennaGain>	float
Range:	-10 to 10
Increment:	0.01
*RST:	0

**Example:** See [Example "To configure data packets"](#) on page 147.**Manual operation:** See ["AntennaX Gain"](#) on page 66**[[:SOURce<hw>]:BB:BT0oth:ECONfig:PCONfig:ANTNumber <AntennaNum>**

Specifies the number of antennas for angle of departure (AoD) direction finding method. You select up to four antennas, that are used for direction finding.

**Parameters:**

<AntennaNum>	integer
Range:	1 to 4
*RST:	1

**Example:** See [Example "To configure data packets"](#) on page 147.**Manual operation:** See ["Antenna Number"](#) on page 66

### 5.7.3 Payload configuration

<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:ACAD.....</a>	156
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:ACAD:APATtern.....</a>	157
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:ACAD:ASELecton.....</a>	157
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:ACID.....</a>	158
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:ACASsigned.....</a>	158
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:SCASsigned.....</a>	158
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:SCID.....</a>	158
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:ICASsigned.....</a>	158
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:ICID.....</a>	158
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:ADID.....</a>	158
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:ALAP.....</a>	159
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECONfiguration:PCONfiguration:ILAP.....</a>	159

## Packet configuration commands - LE

[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SLAP.....	159
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:TLAP.....	159
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:ALENgtH.....	159
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:AMODE.....	160
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:ANUAp.....	160
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:INUAp.....	160
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SNUAp.....	160
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:TNUAp.....	160
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:AOFFset.....	161
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:AOUNits.....	161
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:APHY.....	161
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:ASID.....	162
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:CACCuracy.....	162
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:CID.....	162
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:CINStant.....	163
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:CINTerval.....	163
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:DATA.....	163
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:DATA:DPATtern.....	164
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:DATA:DSELEction.....	164
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:DLENgtH.....	165
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:ECODE.....	165
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:ECOUNTER.....	165
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:EDIVersifier.....	166
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:EHEader:STATe.....	166
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:EHFLags:AADDress:STATe....	166
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:EHFLags:ADINfo:STATe.....	167
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:EHFLags:APTR:STATe.....	167
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:EHFLags:CINFo:STATe.....	167
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:EHFLags:SINFo:STATe.....	167
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:EHFLags:TADDress:STATe....	168
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:EHFLags:TPOWer:STATe.....	168
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:FSBit<ch0>:STATe.....	168
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:FSLength.....	169
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:HLENgtH.....	169
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:LCTimeout.....	169
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MIVector.....	170
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SIVector.....	170
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MROCTets.....	170
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MTOCtets.....	170
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MSKD.....	170
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SSKD.....	170
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MRTTime.....	171
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MTTime.....	171
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MUCHannels.....	171
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MTSPHy:L1M:STATe.....	172
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MTSPHy:L2M:STATe.....	172
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MTSPHy:LCOD:STATe.....	172
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:STMPHy:L1M:STATe.....	172
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:STMPHy:L2M:STATe.....	172
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:STMPHy:LCOD:STATe.....	172

## Packet configuration commands - LE

[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NCINterval.....	172
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NLCTimeout.....	173
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NSLatency.....	173
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NWOFset.....	173
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NWSize.....	174
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:OADJust.....	174
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PHYS:L1M:STATe.....	174
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PHYS:L2M:STATe.....	174
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PHYS:LCOD:STATe.....	174
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:TAType.....	175
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RAType.....	175
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:ROPCode.....	175
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RPHYs:L1M:STATe.....	176
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RPHYs:L2M:STATe.....	176
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RPHYs:LCOD:STATe.....	176
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:TPHYs:L1M:STATe.....	176
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:TPHYs:L2M:STATe.....	176
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:TPHYs:LCOD:STATe.....	176
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RVECtor.....	176
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SCACcuracy.....	177
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SLATency.....	177
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SOUNits.....	177
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SPOFset.....	178
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SVNumber.....	178
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:TPOWER.....	178
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:UTYPE.....	179
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:VNUMber.....	179
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:WOFset.....	180
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:WSIZE.....	180
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:ATYPE.....	180
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:CECount.....	181
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:ID.....	181
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:LPECOUNTER.....	181
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MCLReq.....	181
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PHY:L1M:STATe.....	182
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PHY:L2M:STATe.....	182
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PHY:LCOD:STATe.....	182
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SCECounter.....	183
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SID.....	183

---

**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:ACAD <Data>**

Specifies the pattern source used for additional controller advertising data (ACAD).

**Parameters:**

<Data>                    ALL0 | ALL1 | PATtern | PN09 | PN11 | PN15 | PN16 | PN20 |  
                              PN21 | PN23 | DLISt  
**ALL0 / ALL1**  
 All 0 or all 1 pattern

**PATtern**

User-defined pattern. The pattern can be specified via:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:ACAD:APATtern` on page 157

**PNxx**

Pseudo-random bit sequences (PRBS) of a length of xx bits. The length in bit can be 9, 11, 15, 16, 20, 21, or 23.

**DLIST**

Internal ACAD data list is used. The data list can be specified via:

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:ACAD:ASELecton` on page 157

\*RST: PN09

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["ACAD"](#) on page 77

**[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:ACAD:APATtern**  
**<DPattern>, <BitCount>**

Specifies user-defined pattern. The settings is relevant for

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:ACAD`  
 PATtern

**Parameters:**

**<DPattern>** numeric  
 \*RST: #H0

**<BitCount>** integer  
 Range: 1 to 64  
 \*RST: 1

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["AList / Pattern"](#) on page 78

**[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:ACAD:ASELecton**  
**<DSelection>**

Specifies data list file. The settings is relevant for

`[ :SOURCE<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:ACADDLIST`

**Parameters:**

**<DSelection>** string  
 Path and file name.

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["AList / Pattern"](#) on page 78

**[:SOURCE<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:ACID** <Acid>, <BitCount>

**[:SOURCE<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:ACASsigned** <AcAssigned>, <BitCount>

**[:SOURCE<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:SCASsigned** <ScAssigned>, <BitCount>

**[:SOURCE<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:SCID** <Scid>, <BitCount>

**[:SOURCE<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:ICASsigned** <IcAssigned>, <BitCount>

**[:SOURCE<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:ICID** <Icid>, <BitCount>

Sets the advertiser's device address. For advertising channel packets, the format of the device address differs, depending on the selected address type.

- "Public Address Types"
  - The public address is given from the registration authority IEEE and is composed of:
    - LSB: 24 bits = company\_assigned
    - MSB: 24 bits = company\_id
- "Random Address Type" is a 48-bits random static device address.
- "Private Address Type"
  - A private address is optional and composed of:
    - LSB: 24 bits = hash
    - MSB: 24 bits = random

**Parameters:**

<Icid>	numeric
	*RST: #HACDE48
<BitCount>	integer
	Range: 24 to 24
	*RST: 24

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Manual operation:** See ["Device Address"](#) on page 68

**[:SOURCE<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:ADID** <Adid>, <BitCount>

Specifies "Advertising Data ID" in hexadecimal format to be signaled within an extended header.

**Parameters:**

<Adid> numeric  
 \*RST: #H000

<BitCount> integer  
 Range: 12 to 12  
 \*RST: 12

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["AdvDataInfo Configuration"](#) on page 84

**[ :SOURce<hw> ]:BB:BT0oth:ECOnfiguration:PCONfiguration:ALAP <Lap>, <BitCount>**

**[ :SOURce<hw> ]:BB:BT0oth:ECOnfiguration:PCONfiguration:ILAP <Lap>, <BitCount>**

**[ :SOURce<hw> ]:BB:BT0oth:ECOnfiguration:PCONfiguration:SLAP <Lap>, <BitCount>**

**[ :SOURce<hw> ]:BB:BT0oth:ECOnfiguration:PCONfiguration:TLAP <Lap>, <BitCount>**

Sets the lower address part (LAP) of Bluetooth device address. Commands for the advertising . . . :ALAP, initiating . . . :ILAP, scanning . . . :SLAP PDUs of advertising channel type are provided. In addition, a command is provided for scanner's or initiator's target device address to which the advertisement is directed . . . :TLAP.

**Parameters:**

<Lap> numeric  
 \*RST: #H000080

<BitCount> integer  
 Range: 24 to 24  
 \*RST: 24

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["Device Address"](#) on page 68

**[ :SOURce<hw> ]:BB:BT0oth:ECOnfiguration:PCONfiguration:ALENght <Length>**

Specifies the length of ACAD data pattern.

**Parameters:**

<Length> integer  
 Range: 0 to 62  
 \*RST: 27

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["ACAD Length"](#) on page 77

---

**[:SOURCE<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:AMode <AMode>**

Indicates the mode of the advertisement.

**Parameters:**

<AMode> NCNS | CNS | NCS  
**NCNS:** Non-connectable, non-scannable  
**CNS:** Connectable, non-scannable  
**NCS:** Non-connectable, non-scannable  
 \*RST: NCNS

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["Advertising Mode"](#) on page 75

---

**[:SOURCE<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:ANUap <NapUap>, <BitCount>**

**[:SOURCE<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:INUap <NapUap>, <BitCount>**

**[:SOURCE<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:SNUap <NapUap>, <BitCount>**

**[:SOURCE<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:TNUap <NapUap>, <BitCount>**

Sets the non-significant address part (NAP) and upper address part (UAP) of Bluetooth device address. Commands for the advertising . . . :ANUap, initiating . . . :INUap, and scanning . . . :SNUap PDUs of advertising channel type are provided. In addition, a command is provided for scanner's or initiator's target device address to which the advertisement is directed . . . :TNUap.

**Parameters:**

<NapUap> numeric  
 \*RST: #HACDE48  
 <BitCount> integer  
 Range: 24 to 24  
 \*RST: 24

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["Device Address"](#) on page 68



---

**[ :SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:AOffSet <AOffset>**

Specifies the time from the start of the packet containing the AuxPtr field to the approximate start of the auxiliary packet. The offset is determined by multiplying the value by the unit, see

`[ :SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:AOUNits`

**Parameters:**

<AOffset> float  
 Range: 0 to 245.7 or 246 to 2457 depending on offset unit

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["Aux Offset"](#) on page 85

---

**[ :SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:AOUNits <Unit>**

Indicates the units used by the "Aux Offset" parameter, see

`[ :SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:AOffSet`

**Parameters:**

<Unit> U30 | U300  
**U30:** 30 µs  
**U300:** 300 µs  
 \*RST: U30

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["Offset Units"](#) on page 85

---

**[ :SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:APHY <APhy>**

Specifies the physical layer used to transmit the auxiliary packet.

**Parameters:**

<APhy> L1M | L2M | LCOd  
 LE 1M, LE 2M, LE coded PHY  
 \*RST: L1M

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["AUX PHY"](#) on page 85

---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:ASID <Asid>, <BitCount>**

Specifies the "Advertising Set ID" in hexadecimal format to be signaled within an extended header.

**Parameters:**

<Asid>                    numeric  
                           \*RST:        #H0

<BitCount>               integer  
                           Range:        4 to 4  
                           \*RST:        4

**Example:**                See [Example "To configure advertising packets"](#) on page 144.

**Options:**                R&S SMM-K117

**Manual operation:**    See ["AdvDataInfo Configuration"](#) on page 84

---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CACCuracy <CAccuracy>**

Specifies the clock accuracy of the advertiser used between the packet containing this data and the auxiliary packet.

**Parameters:**

<CAccuracy>               T500 | T50  
                           **T500:** 51 ppm to 500 ppm  
                           **T50:** 0 ppm to 50 ppm  
                           \*RST:        T500

**Example:**                See [Example "To configure advertising packets"](#) on page 144.

**Options:**                R&S SMM-K117

**Manual operation:**    See ["Clock Accuracy"](#) on page 85

---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CID <Cid>, <BitCount>**

Sets the company identifier of the manufacturer of the Bluetooth Controller. A 16 bit value is set.

**Note:** This parameter is relevant for data frame configuration and for the packet type LL\_VERSION\_IND.

**Parameters:**

<Cid>                      numeric  
                           \*RST:        0

<BitCount> integer  
 Range: 16 to 16  
 \*RST: 16

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Company Id\(hex\)"](#) on page 75

**[ :SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CINStant**  
 <CInstant>

Sets a connection instant for indicating the connection event at which the new connection parameters are taken in use.

**Parameters:**

<CInstant> integer  
 Range: 1 to depends on sequence length  
 \*RST: 1

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Connection Instant"](#) on page 72

**[ :SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CINterval**  
 <CInterval>

Sets the time interval between the start points of two consecutive connection events for the packet type DATA and all CONTROL\_DATA packet types.

Command sets the values in ms. Query returns values in s.

**Parameters:**

<CInterval> float  
 Range: 7.5E-3 s to depends on oversampling  
 Increment: 1.25E-3 s  
 \*RST: 7.5E-3 s  
 Default unit: ms

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Connection Event Interval"](#) on page 59

**[ :SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:DATA** <Data>

Selects the pattern source used for the payload.

**Parameters:**

<Data> ALL0 | ALL1 | PATtern | PN09 | PN11 | PN15 | PN16 | PN20 |  
 PN21 | PN23 | DLISt

**ALL0 / ALL1**

All 0 or all 1 pattern

**PATtern**

User-defined pattern. The pattern can be specified via:

`[ :SOURce<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:DATA:DPATtern` on page 164

**PNxx**

Pseudo-random bit sequences (PRBS) of a length of xx bits. The length in bit can be 9, 11, 15, 16, 20, 21, or 23.

**DLISt**

Internal data list is used. The data list can be specified via:

`[ :SOURce<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:DATA:DSELection` on page 164

\*RST: PN09

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Manual operation:** See ["Data Source"](#) on page 70

`[ :SOURce<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:DATA:DPATtern  
<DPattern>, <BitCount>`

Specifies the user-defined pattern. The setting is relevant for

`[ :SOURce<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:DATA  
PATtern`

**Parameters:**

<DPattern>	numeric
*RST:	#H0
<BitCount>	integer
Range:	1 to 64
*RST:	1

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Data Source"](#) on page 70

`[ :SOURce<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:DATA:DSELection  
<DSelection>`

Specifies data list file. The setting is relevant for

`[ :SOURce<hw> ] :BB:BT0oth:ECOnfiguration:PCOnfiguration:DATA:DLISt`

**Parameters:**

<DSelection>	string
--------------	--------

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Manual operation:** See ["Data Source"](#) on page 70

---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:DLEnGth**  
<DLength>

Sets the payload data length in bytes.

**Parameters:**

<DLength>                    integer  
                                   Range:     0 to 255 (advertiser) or 251 (data)  
                                   \*RST:     31

**Example:**                    See [Example "To configure advertising packets"](#) on page 144.

**Manual operation:**        See ["Data Length"](#) on page 71

---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:ECODE** <ECode>,  
<BitCount>

Sets the error code value to inform the remote device why the connection is about to be terminated in case of LL\_TERMINATE\_IND packet. On the other hand, this parameter for LL\_REJECT\_IND packet is used for the reason a request was rejected. A 8 bit value is set.

**Note:** This parameter is relevant for data frame configuration and the packet type:

- LL\_TERMINATE\_IND
- LL\_REJECT\_IND

**Parameters:**

<ECode>                    numeric  
                                   \*RST:     #H00

<BitCount>                integer  
                                   Range:     8 to 8  
                                   \*RST:     8

**Example:**                    SOURce1:BB:BT0:ECOn:PCOn:ECOD #H00,8  
                                   Sets the error code.

**Manual operation:**        See ["Error Code"](#) on page 74

---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:ECOUNTER**  
<ECounter>

Counts the AUX\_SYNC\_IND packets that the SyncInfo field describes.

**Parameters:**

<ECounter>                integer  
                                   Range:     0 to 65535  
                                   \*RST:     0

**Example:**                    See [Example "To configure advertising packets"](#) on page 144.

**Options:**                    R&S SMM-K117

**Manual operation:** See ["Event Counter"](#) on page 86

---

**[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:EDIVersifier**  
 <EDiversifier>, <BitCount>

Sets the encrypted diversifier of the Central for device identification. The parameter is an initialization vector provided by the host in the HCI\_ULP\_Start\_Encryption command.

**Parameters:**

<EDiversifier>	numeric
	*RST: #H0000
<BitCount>	integer
	Range: 16 to 16
	*RST: 16

**Example:** SOURCE1:BB:BTO:ECON:PCON:EDIV #H0000,16  
 Sets the encrypted diversifier of the Central.

**Manual operation:** See ["Encrypted DIVersifier \(hex\)"](#) on page 73

---

**[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:EHEader:STATE**  
 <State>

Enables / disables extended header for advertising packets with scanning PDUs.

**Parameters:**

<State>	1   ON   0   OFF
	*RST: 1

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["Extended Header"](#) on page 75

---

**[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:EHFLags:**  
**AADDRESS:STATE** <State>

If enabled, the R&S SMM100A includes the signaling of non-significant advertising address part (NAP) and upper address part (UAP).

**Parameters:**

<State>	1   ON   0   OFF
	*RST: 0

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["AdvA"](#) on page 76

---

**[[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:EHFLags:ADINfo: STATE <State>**

Enables / disables the signaling of advertising data information consisting of "Advertising Data ID" and "Advertising Set ID".

**Parameters:**

<State>                    1 | ON | 0 | OFF  
 \*RST:                    0

**Example:**                See [Example "To configure advertising packets"](#) on page 144.

**Options:**                R&S SMM-K117

**Manual operation:**    See ["AdvData Info"](#) on page 76

---

**[[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:EHFLags:APTR: STATE <State>**

Enables / disables secondary advertising channel.

**Parameters:**

<State>                    1 | ON | 0 | OFF  
 \*RST:                    0

**Example:**                See [Example "To configure advertising packets"](#) on page 144.

**Options:**                R&S SMM-K117

**Manual operation:**    See ["AuxPtr"](#) on page 77

---

**[[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:EHFLags:CINfo: STATE <State>**

Activates the CTEInfo field in the extended header of Bluetooth LE advertising packets in the LE uncoded PHY.

**Parameters:**

<State>                    1 | ON | 0 | OFF  
 \*RST:                    0

**Example:**                See [Example "To configure advertising packets"](#) on page 144.

**Manual operation:**    See ["CTE Info"](#) on page 76

---

**[[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:EHFLags:SINfo: STATE <State>**

Enables / disables the signaling of SynclInfo field for periodic advertisement.

**Parameters:**

<State>                    1 | ON | 0 | OFF  
 \*RST:                    0

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["SyncInfo"](#) on page 77

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:EHFLags: TADdress:STATe <State>**

Enables / disables the signaling of non-significant address part (NAP) and upper address part (UAP) of a target address.

**Parameters:**

<State> 1 | ON | 0 | OFF  
\*RST: 0

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["TargetA"](#) on page 76

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:EHFLags:TPOWER: STATE <State>**

Enables the signaling of required transmit power.

**Parameters:**

<State> 1 | ON | 0 | OFF  
\*RST: 0

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["TxPow"](#) on page 77

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:FSBit<ch0>:STATe <State>**

Enables / disables features, related to bit numbers 0 to 27 for the used feature set. See also [Table 3-8](#).

Information is transmitted via LL\_FEATURE\_REQ, LL\_FEATURE\_RSP, LL\_PERIPHERAL\_FEATURE\_REQ.

**Parameters:**

<State> 1 | ON | 0 | OFF  
\*RST: 0

**Example:** See [Example "To configure data packets"](#) on page 147.

**Options:** R&S SMM-K117

**Manual operation:** See ["FeatureSet Configuration"](#) on page 87



---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:FSLength**  
<FsLength>

Enables that the feature set length is indicated.

**Parameters:**

<FsLength>            integer  
                          Range:     1 to 26  
                          \*RST:     8

**Example:**            See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Feature Set Length"](#) on page 73

---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:HLEnGth**  
<HLength>

Requires data event and advertising frame configuration with the packet type CONNECT\_IND.

Sets the difference from the current channel to the next channel.

The Central and Peripherals determine the data channel in use for every connection event from the channel map. Hop\_length is set for the LL connection and communicated in the CONNECT\_IND and LL\_CHANNEL\_MAP\_IND packets.

**Parameters:**

<HLength>            integer  
                          Range:     5 to 16  
                          \*RST:     5

**Example:**            See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Hop Length"](#) on page 72

---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:LCTimeout**  
<LcTimeout>

Defines the maximum time between two correctly received Bluetooth LE packets in the LL connection before the connection is considered lost for the packet type CONNECT\_IND.

Command sets the values in ms. Query returns values in s.

**Parameters:**

<LcTimeout>           float  
                          Range:     100E-3 s to 32000E-3 s  
                          Increment: 10E-3 s  
                          \*RST:     100E-3 s  
                          Default unit: ms

**Example:**            See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["LL Connection Timeout"](#) on page 71

---

```
[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:MIVector
  <MiVector>, <BitCount>
```

```
[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:SIVector
  <SiVector>, <BitCount>
```

Sets the portion of Central or the portion of the Peripheral of the initialization vector (IVm/IVs).

**Parameters:**

```
<SiVector>          numeric
                    *RST:    #H0

<BitCount>         integer
                    Range:   32 to 32
                    *RST:   32
```

**Example:**

```
SOURCE1:BB:BTO:ECON:PCON:MIV
#H0000000000000000,32
(Central).
SOURCE1:BB:BTO:ECON:PCON:SIV
#H0000000000000000,32
(Peripheral).
```

---

```
[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:MROctets
  <MROctets>
```

```
[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:MTOctets
  <MTOctets>
```

Specifies the maximum allowed payload length of a packet to be received (. . :MROctets) or transmitted (. . :MTOctets). Information is signaled via LL\_LENGTH\_REQ and LL\_LENGTH\_RSP.

**Parameters:**

```
<MTOctets>         integer
                    Range:   27 to 251
                    *RST:   27
```

**Example:** See [Example "To configure data packets"](#) on page 147.

**Options:** R&S SMM-K117

**Manual operation:** See ["Max Rx Octets / Max Tx Octets"](#) on page 79

---

```
[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:MSKD <Mskd>,
  <BitCount>
```

```
[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:SSKD <Sskd>,
  <BitCount>
```

Sets the portion of Central or the portion of the Peripheral of the session key diversifier (SKDm/SKDs).

**Parameters:**

<Sskd> numeric  
 \*RST: #H0

<BitCount> integer  
 Range: 64 to 64  
 \*RST: 64

**Example:**

SOURce1:BB:BTO:ECON:PCON:MSKD  
 #H0000000000000000,64  
 (Central).  
 SOURce1:BB:BTO:ECON:PCON:SSKD  
 #H0000000000000000,64  
 (Peripheral).

---

**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MRTime <MRTime>**  
**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MTTime <MTTime>**

Specifies the maximum allowed time to receive (. . :MRTime) or transmit (. . :MTTime) a packet. Information is signaled via LL\_LENGTH\_REQ and LL\_LENGTH\_RSP.

**Parameters:**

<MTTime> float  
 Range: 0.328E-3 to 17.04E-3  
 Increment: 0.001E-3  
 \*RST: 17.04E-3

**Example:** See [Example "To configure data packets"](#) on page 147.

**Options:** R&S SMM-K117

**Manual operation:** See ["Max Rx Time / Max Tx Time"](#) on page 79

---

**[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MUCHannels**  
 <Muchannels>

Specifies the minimum number of channels to be used on the specified PHYs, see

[\[:SOURce<hw>\]:BB:BT0oth:ECONfiguration:PCONfiguration:PHYS:L1M:STATE](#) etc.

**Parameters:**

<Muchannels> integer  
 Range: 2 to 37  
 \*RST: 2

**Example:** See [Example "To configure data packets"](#) on page 147.

**Options:** R&S SMM-K117

**Manual operation:** See ["Min Used Channels"](#) on page 81

---

```
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MTSPHy:L1M:
  STATE <MTSP>
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MTSPHy:L2M:
  STATE <MTSP>
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MTSPHy:LCOD:
  STATE <MTSP>
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:STMPHy:L1M:
  STATE <STMP>
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:STMPHy:L2M:
  STATE <STMP>
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:STMPHy:LCOD:
  STATE <STMP>
```

Specifies the physical layers in Central-to-Peripheral (. . . :MTSPHy: . . .) or Peripheral-to-Central (. . . :STMPHy: . . .) direction. Information is signaled via LL\_PHY\_UPDATE\_IND.

You can enable one or more PHYs:

- L1M: for LE uncoded 1 Msymbol/s PHY.
- L2M: for LE uncoded 2 Msymbol/s PHY.
- LCOD: for LE coded 1 Msymbol/s PHY.

**Parameters:**

```
<STMP>          1 | ON | 0 | OFF
                *RST:      0
```

**Example:** See [Example "To configure data packets"](#) on page 147.

**Options:** R&S SMM-K117

**Manual operation:** See ["M\\_TO\\_S\\_PHY/S\\_TO\\_M\\_PHY"](#) on page 80

---

```
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NCINterval
  <NcInterval>
```

Sets the time interval new connection events for the packet types CONNECT\_IND and LL\_CONNECTION\_UPDATE\_IND.

Command sets the values in ms. Query returns values in s.

**Parameters:**

```
<NcInterval>    float
                Range:      7.5E-3 s to depends on oversampling
                Increment:  1.25E-3 s
                *RST:      7.5E-3 s
                Default unit: ms
```

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Connection Evt Interval"](#) on page 71



**Parameters:**

<NwOffset> float  
 Range: 0 s to depends on connection event interval  
 Increment: 1.25E-3 s  
 \*RST: 0  
 Default unit: ms

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Transmit Window Offset"](#) on page 74

**[ :SOURCE<hw> ]:BB:BT0oth:ECOnfiguration:PCONfiguration:NWSize <NwSize>**

Sets the size of the transmit window, regarding to the start point for data event and advertising frame configuration with the packet type LL\_CONNECTION\_UPDATE\_IND.

**Parameters:**

<NwSize> float  
 Range: 1.25E-3 to depends on connection event interval  
 Increment: 1.25E-3  
 \*RST: 1.25E-3

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Transmit Window Size"](#) on page 73

**[ :SOURCE<hw> ]:BB:BT0oth:ECOnfiguration:PCONfiguration:OADJust <State>**

Adjusts the "Sync Packet Offset" automatically to the next value, which is a multiple of the "Offset Units".

**Parameters:**

<State> 1 | ON | 0 | OFF  
 \*RST: 0

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Offset Adjust"](#) on page 86

**[ :SOURCE<hw> ]:BB:BT0oth:ECOnfiguration:PCONfiguration:PHYS:L1M:STATE <State>**

**[ :SOURCE<hw> ]:BB:BT0oth:ECOnfiguration:PCONfiguration:PHYS:L2M:STATE <State>**

**[ :SOURCE<hw> ]:BB:BT0oth:ECOnfiguration:PCONfiguration:PHYS:LCOD:STATE <State>**

Specifies the physical layers for which the Peripheral has a minimum number of used channels requirement. Information is signaled via LL\_MIN\_USED\_CHANNELS\_IND.

You can enable one or more PHYs:

- L1M: for LE uncoded 1 Msymbol/s PHY.

- L2M: for LE uncoded 2 Msymbol/s PHY.
- LCOD: for LE coded 1 Msymbol/s PHY.

**Parameters:**

<State> 1 | ON | 0 | OFF  
 \*RST: 0

**Example:** See [Example "To configure data packets"](#) on page 147.

**Options:** R&S SMM-K117

**Manual operation:** See ["PHYs"](#) on page 80

**[:SOURCE<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:TAType <TaType>  
 [:SOURCE<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RAType <RaType>**

Selects the address type of the controller device.

Depending on the Bluetooth controller role either Tx or Rx or both address types are assigned. Subdivided into private and random, a Bluetooth LE device address consists of 48 bits. The format of the device address differs depending on the selected address type.

**Parameters:**

<RaType> PUBLIC | RANDom

**PUBLIC**

Allocates a unique 48 bit address to each Bluetooth LE device. The public address is given from the registration authority IEEE.

**RANDom**

Allocates a 48-bit address to each Bluetooth LE device. A random address is optional.

\*RST: PUBLIC

**Example:**

SOURCE1:BB:BT0:ECON:PCON:TATY PUBL  
 SOURCE1:BB:BT0:ECON:PCON:RATY RAND

**[:SOURCE<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:ROPCode  
 <ROpcode>, <BitCount>**

Specifies the Opcode of rejected LL control PDU. information is signaled via LL\_REJECT\_EXT\_IND.

**Parameters:**

<ROpcode> numeric  
 \*RST: #H00

<BitCount> integer  
 Range: 8 to 8  
 \*RST: 8

**Example:** See [Example "To configure data packets"](#) on page 147.

**Options:** R&S SMM-K117

**Manual operation:** See ["Reject Opcode"](#) on page 80

---

```
[ :SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RPHYs:L1M:STATe
<RPhys>
```

```
[ :SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RPHYs:L2M:STATe
<RPhys>
```

```
[ :SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RPHYs:LCOD:
STATe <RPhys>
```

```
[ :SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:TPHYs:L1M:STATe
<TPHys>
```

```
[ :SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:TPHYs:L2M:STATe
<TPHys>
```

```
[ :SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:TPHYs:LCOD:
STATe <TPHys>
```

Specifies preferred physical layers in Rx ( . . . :RPHYs: . . . ) or Tx ( . . . :TPHYs: . . . ) direction. Information is signaled via LL\_PHY\_REQ and LL\_PHY\_RSP.

You can enable one or more PHYs: :L1M: for LE uncoded 1 Msymbol/s PHY, :L2M: for LE uncoded 2 Msymbol/s PHY, and :LCOD: for LE coded 1 Msymbol/s PHY.

**Parameters:**

```
<TPHys>          1 | ON | 0 | OFF
*RST:            0
```

**Example:** See [Example "To configure data packets"](#) on page 147.

**Options:** R&S SMM-K117

**Manual operation:** See ["Rx PHY / Tx PHY"](#) on page 79

---

```
[ :SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RVEctor
<RVector>, <BitCount>
```

Sets the random vector of the Central for device identification.

The parameter is an initialization vector provided by the Host in the HCI\_ULP\_Start\_Encryption command.

**Parameters:**

```
<RVector>        numeric
*RST:            #H0

<BitCount>       integer
Range:          64 to 64
*RST:           64
```

**Example:** SOURce1:BB:BT0:ECON:PCON:RVEC  
#H000000000000000000,64  
Sets the random vector of the Central.

**Manual operation:** See ["Random Vector \(hex\)"](#) on page 72



---

**[ :SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SCACcuracy**  
 <ScAccuracy>

Defines the clock accuracy of the Central with specified encoding.

This parameter is used by the Peripheral to determine required listening windows in the LL connection. It is a controller design parameter known by the bluetooth controller.

**Parameters:**

<ScAccuracy>            SCA0 | SCA1 | SCA2 | SCA3 | SCA4 | SCA5 | SCA6 | SCA7  
 \*RST:                    SCA0

**Example:**

SOURce1:BB:BT0:ECON:PCON:SCAC SCA1  
 Sets the encoding value.

**Manual operation:** See "[Sleep Clock Accuracy](#)" on page 74

---

**[ :SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SLATency**  
 <SLatency>

Requires data event and advertising frame configuration with the packet type CONNECT\_IND.

Sets the number of consecutive connection events the Peripheral can ignore for asymmetric link layer connections.

**Parameters:**

<SLatency>            integer  
 Range:                0 to depends on LL connection timeout and connection event interval  
 \*RST:                 depends on LL connection timeout and connection event interval

**Example:**

SOURce1:BB:BT0:UPTY CREQ  
 Sets packet type CONNECT\_IND.  
 SOURce1:BB:BT0:ECON:PCON:SLAT 10  
 Sets the number of consecutive connection events.

**Manual operation:** See "[Peripheral Latency](#)" on page 71

---

**[ :SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SOUNits <Unit>**

Indicates the units used by the "Sync Packet Offset" parameter, see

[\[:SOURce<hw>\]:BB:BT0oth:ECONfiguration:PCONfiguration:SPOffset](#)

**Parameters:**

<Unit>                    U30 | U300  
**U30**  
 30 µs  
**U300**  
 300 µs  
 \*RST:                    U30

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["Offset Units"](#) on page 86

**[ :SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:SPOffset**  
 <SPOffset>

Specifies the time from the start of the AUX\_ADV\_IND packet containing the SyncInfo field to the start of the AUX\_SYNC\_IND packet. The offset is determined by multiplying the value by the unit, see

[\[:SOURce<hw>\]:BB:BT0oth:ECOnfiguration:PCONfiguration:SOUNits](#)

**Parameters:**

<SPOffset> float  
 Range: 0 to 245.7 or 246 to 2457 depending on offset unit

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["Sync Packet Offset"](#) on page 86

**[ :SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:SVNumber**  
 <SvNumber>, <BitCount>

Sets a unique value for each implementation or revision of an implementation of the Bluetooth Controller. A 16-bit value is set.

**Note:** This parameter is relevant for data frame configuration and for the packet type: LL\_VERSION\_IND.

**Parameters:**

<SvNumber> numeric  
 \*RST: 0

<BitCount> integer  
 Range: 16 to 16  
 \*RST: 16

**Example:** SOURCE1:BB:BT0:ECOn:PCON:SVN #H0000,16  
 Sets the sub version number.

**Manual operation:** See ["SubVersion Number\(hex\)"](#) on page 75

**[ :SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:TPOWER** <TPower>

Sets the required transmit power to be signaled within an extended header.

**Parameters:**

<TPower> integer  
 Range: -127 to 126  
 \*RST: 0

**Example:** See [Example "To configure advertising packets"](#) on page 144.

**Options:** R&S SMM-K117

**Manual operation:** See ["TxPow"](#) on page 77

**[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:UTYPE <UType>, <BitCount>**

Enables that an invalid control packet is indicated.

The CtrType field indicates the value of the LL control packet that caused the transmission of this packet.

**Parameters:**

<UType> numeric  
 \*RST: #H0

<BitCount> integer  
 Range: 8 to 8  
 \*RST: 8

**Example:** SOURCE1:BB:BTO:ECON:PCON:UTYP #H00,8  
 Enables that an invalid control packet is indicated.

**Manual operation:** See ["Unknown Type \(hex\)"](#) on page 71

**[:SOURCE<hw>]:BB:BTOoth:ECONfiguration:PCONfiguration:VNUMBER <VNumber>, <BitCount>**

Sets the company identifier of the manufacturer of the Bluetooth controller. An 8-bit value is set.

**Note:** This parameter is relevant for data frame configuration and for the packet type LL\_VERSION\_IND.

**Parameters:**

<VNumber> numeric  
 \*RST: 0

<BitCount> integer  
 Range: 8 to 8  
 \*RST: 8

**Example:** SOURCE1:BB:BTO:ECON:PCON:VNUM #H00,8  
 Sets the version number.

**Manual operation:** See ["Version Number\(hex\)"](#) on page 75

---

**[[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:WOffset  
<WOffset>**

Sets the start point of the window transmit for data event and advertising frame configuration with the packet type CONNECT\_IND.

Command sets the values in ms. Query returns values in s.

**Parameters:**

<WOffset> float  
 Range: 0 s to depending on connection event interval  
 Increment: 1.25E-3 s  
 \*RST: 0 s  
 Default unit: ms

**Example:** See [Example "Configure event and frame configuration settings"](#) on page 134.

**Manual operation:** See ["Transmit Window Offset"](#) on page 57  
 See ["Transmit Window Offset"](#) on page 74

---

**[[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:WSize <WSize>**

Sets the size of the transmit window, regarding to the start point for data event and advertising frame configuration with the packet type CONNECT\_IND.

**Parameters:**

<WSize> float  
 Range: 1.25E-3 to depends on connection event interval  
 Increment: 1.25E-3  
 \*RST: 1.25E-3

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Transmit Window Size"](#) on page 73

---

**[[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:ATYPe <AType>**

Sets the address type in the payload of Bluetooth LE LL\_PERIODIC\_SYNC\_IND packets.

**Parameters:**

<AType> PUBLIC | RANDom  
 \*RST: PUBLIC

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Address Type"](#) on page 82

---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CECount**  
 <CECount>

Specifies the connection event count in the CtrData field of the LL\_PERIODIC\_SYNC\_IND control data PDU.

**Parameters:**

<CECount>            integer  
                          Range:     0 to 65535  
                          \*RST:     0

**Example:**            See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Connection Event Count"](#) on page 81

---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:ID** <Id>, <BitCount>

Specifies the ID in the CtrData field of the LL\_PERIODIC\_SYNC\_IND PDU.

**Parameters:**

<Id>                    numeric  
                          \*RST:     #HAAAA

<BitCount>            integer  
                          Range:     16 to 16  
                          \*RST:     16

**Example:**            See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["ID\(hex\)"](#) on page 81

---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:LPECOUNTER**  
 <LPECOUNTER>

Specifies the lastPaEventCounter field in the CtrData field of the LL\_PERIODIC\_SYNC\_IND PDU.

**Parameters:**

<LPECOUNTER>        integer  
                          Range:     0 to 65535  
                          \*RST:     0

**Example:**            See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Last Pa Event Counter"](#) on page 81

---

**[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:MCLReq**  
 <MCLReq>

Specifies the minimum CTE length in the CtrData field of the LL\_CTE\_Req PDU.

**Parameters:**

<MCLReq> float  
 Range: 16E-6 to 160E-6  
 Increment: 8E-6  
 \*RST: 16E-6

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["MinCTELenReq"](#) on page 83  
 See ["CTETypeReq"](#) on page 83

**[ :SOURCE<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PHY:L1M:STATe**  
 <State>

Sets the LE 1M PHY in the CtrData field of the LL\_PERIODIC\_SYNC\_IND PDU.

**Parameters:**

<State> 1 | ON | 0 | OFF  
 \*RST: 0

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["PHY"](#) on page 82

**[ :SOURCE<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PHY:L2M:STATe**  
 <State>

Sets the LE 2M PHY in the CtrData field of the LL\_PERIODIC\_SYNC\_IND PDU.

**Parameters:**

<State> 1 | ON | 0 | OFF  
 \*RST: 0

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["PHY"](#) on page 82

**[ :SOURCE<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PHY:LCOD:STATe**  
 <State>

Sets the LE Coded PHY in the CtrData field of the LL\_PERIODIC\_SYNC\_IND PDU.

**Parameters:**

<State> 1 | ON | 0 | OFF  
 \*RST: 0

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["PHY"](#) on page 82

---

```
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SCECounter
<SCECounter>
```

**Parameters:**

```
<SCECounter>      integer
                   Range:    0 to 65535
                   *RST:     0
```

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["Sync Connection Event Counter"](#) on page 83

---

```
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SID <Sid>,
<BitCount>
```

Specifies the SID in the CtrData field of the LL\_PERIODIC\_SYNC\_IND.

**Parameters:**

```
<Sid>              numeric
                   *RST:    #H0

<BitCount>        integer
                   Range:    4 to 4
                   *RST:     4
```

**Example:** See [Example "To configure data packets"](#) on page 147.

**Manual operation:** See ["SID\(hex\)"](#) on page 82

## 5.8 Test packet configuration commands - LE

**Example: To configure test packets**

```
// *****
// Set pattern, payload length, time interval.
// *****
SOURcel:BB:BT0oth:CTYPe DATA
SOURcel:BB:BT0oth:UPTYPe TPAC
SOURcel:BB:BT0oth:DTTest:TPConfiguration:TPINterval 12.5
SOURcel:BB:BT0oth:DTTest:TPConfiguration:UPSource PAT3
SOURcel:BB:BT0oth:DTTest:TPConfiguration:UPLength 255

/ *****
// Alternatively select packet type LE coded and set its coding.
// *****
SOURcel:BB:BT0oth:PFORMAT LCOD
SOURcel:BB:BT0oth:ECONfiguration:PCONfiguration:SPBit TWO
```

<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:ECOnfiguration:PCOnfiguration:SPBit.....</a>	184
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:TPConfiguration:TPInterval.....</a>	184
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:TPConfiguration:UPLength.....</a>	184
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:DTTest:TPConfiguration:UPSource.....</a>	185

---

### **`[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCOnfiguration:SPBit <SPB>`**

Specifies a coding for LE coded packets. The specification for Bluetooth wireless technology defines two values S for forward error correction: S = 2 symbol/bit and S = 8 symbol/bit.

#### **Parameters:**

`<SPB>` TWO | EIGHT  
 \*RST: TWO

**Example:** See [Example "To configure test packets"](#) on page 183.

**Options:** R&S SMM-K117

**Manual operation:** See ["Symbols per a Bit"](#) on page 89

---

### **`[:SOURce<hw>]:BB:BT0oth:DTTest:TPConfiguration:TPInterval <TpInterval>`**

Sets the time interval between two consecutive test packets, regarding the starting points.

Command sets the values in ms. Query returns values in s.

#### **Parameters:**

`<TpInterval>` float  
 Range: 0.625E-3 s to 27.5E-3 s - depends on packet characteristics  
 Increment: 0.625E-3 s  
 \*RST: 0.625E-3 s  
 Default unit: ms

**Example:** See [Example "To configure test packets"](#) on page 183.

**Manual operation:** See ["Packet Interval"](#) on page 88

---

### **`[:SOURce<hw>]:BB:BT0oth:DTTest:TPConfiguration:UPLength <UpLength>`**

Sets the payload length.

#### **Parameters:**

`<UpLength>` integer  
 Range: 0 to 255  
 \*RST: 37

**Example:** See [Example "To configure test packets"](#) on page 183.

**Manual operation:** See ["Payload Length"](#) on page 89



**[[:SOURce<hw>]:BB:BT0oth:DTTest:TPConfiguration:UPSource <UpSource>**

Selects the data source used for the payload test packets.

**Parameters:**

<UpSource> PN09 | PAT1 | PAT2 | PN15 | PAT3 | PAT4 | PAT5 | PAT6

**PN9 / PN15**  
Pseudo-random bit sequences (PRBS) of a length of xx bits.  
The length in bit can be 9 or 15.

**PAT1**  
Predefined pattern: 11110000

**PAT2**  
Predefined pattern: 10101010

**PAT3**  
Predefined pattern: 11111111

**PAT4**  
Predefined pattern: 00000000

**PAT5**  
Predefined pattern: 00001111

**PAT6**  
Predefined pattern: 01010101

\*RST: PN09

**Example:** See [Example "To configure test packets"](#) on page 183.

**Manual operation:** See ["Payload Type"](#) on page 89

## 5.9 Filter/clipping commands

<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:CLIPping:LEVel</a> .....	186
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:CLIPping:MODE</a> .....	186
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:CLIPping:STATE</a> .....	186
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:FILTer:TYPE</a> .....	187
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:FILTer:ILENght</a> .....	187
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:FILTer:ILENght:AUTO[:STATE]</a> .....	187
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:FILTer:OSAMpling</a> .....	188
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:FILTer:OSAMpling:AUTO[:STATE]</a> .....	188
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:FILTer:MINDEX</a> .....	188
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:FILTer:MTYPE</a> .....	188
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:MSETtings:FDEVIation</a> .....	189
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:FILTer:PARAMeter:APCO25</a> .....	189
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:FILTer:PARAMeter:COSine</a> .....	189
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:FILTer:PARAMeter:FGAuss</a> .....	189
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:FILTer:PARAMeter:GAUSS</a> .....	190
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:FILTer:PARAMeter:LPASS</a> .....	190
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:FILTer:PARAMeter:PGAuss</a> .....	190

<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:FILTer:PARAmeter:RCOSine.....</a>	191
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:FILTer:PARAmeter:SPHase.....</a>	191
<a href="#">[:SOURce&lt;hw&gt;]:BB:BT0oth:SRATe:VARiAtion.....</a>	191

---

### **[:SOURce<hw>]:BB:BT0oth:CLIPping:LEVel <Level>**

Sets the limit for level clipping.

This value indicates at what point the signal is clipped. It is specified as a percentage, relative to the highest level. 100% indicates that clipping does not take place.

#### **Parameters:**

<Level>                    integer  
                               Range:     1 to 100  
                               Increment: 1  
                               \*RST:     100  
                               Default unit: PCT

#### **Example:**

```
SOURce1:BB:BTO:CLIP:LEV 80
Sets the limit for level clipping to 80% of the maximum level.
SOURce1:BB:BTO:CLIP:STAT ON
Activates level clipping.
```

**Manual operation:** See "[Clipping Level](#)" on page 95

---

### **[:SOURce<hw>]:BB:BT0oth:CLIPping:MODE <Mode>**

The command sets the method for level clipping.

#### **Parameters:**

<Mode>                    VECTor | SCALar  
**VECTor**  
 The reference level is the amplitude  $|i+jq|$ .  
**SCALar**  
 The reference level is the absolute maximum of the I and Q values.  
 \*RST:                    VECTor

#### **Example:**

```
SOURce1:BB:BTO:CLIP:MODE VECT
Sets the amplitude as reference level.
```

**Manual operation:** See "[Clipping Mode](#)" on page 95

---

### **[:SOURce<hw>]:BB:BT0oth:CLIPping:STATe <State>**

Activates level clipping.

The value is defined with the command `BB:BTO:CLIPping:LEVel`, the mode of calculation with the command `BB:BTO:CLIPping:MODE`.

**Parameters:**

<State> 1 | ON | 0 | OFF  
 \*RST: OFF

**Example:**

SOURce1:BB:BTO:CLIP:STAT ON  
 Activates level clipping.

**Manual operation:** See "[Clipping State](#)" on page 95

**[:SOURce<hw>]:BB:BTOoth:FILTer:TYPE <Type>**

Selects the filters used for  $\pi/4$  DQPSK and 8DPSK modulations. This opens a selection window containing all the filters available to the instrument.

**Parameters:**

<Type> RCOSine | COSine | GAUSs | LGAuss | CONE | COF705 |  
 COEQUALizer | COFEQUALizer | C2K3x | APCO25 | SPHase |  
 RECTangle | PGAuss | LPASs | DIRac | ENPShape |  
 EWPSshape  
 \*RST: RCOSine

**Example:**

SOURce1:BB:BTO:FILT:TYPE RCOS  
 Sets the filter type RCOSine.

**Manual operation:** See "[Filter](#)" on page 92

**[:SOURce<hw>]:BB:BTOoth:FILTer:ILENgtH <Length>**

Sets the impulse length (the number of filter taps).

**Parameters:**

<Length> integer  
 Range: 1 to depends on oversampling  
 \*RST: 32

**Example:**

SOURce1:BB:BTO:FILT:ILEN 10  
 Sets the number of filter tabs to 10.

**Manual operation:** See "[Impulse Length](#)" on page 92

**[:SOURce<hw>]:BB:BTOoth:FILTer:ILENgtH:AUTO[::STATe] <State>**

Activates the impulse length state.

If activated, the most sensible parameter values are selected. The value depends on the coherence check.

**Parameters:**

<State> 1 | ON | 0 | OFF  
 \*RST: 1

**Example:**

SOURce1:BB:BTO:FILT:ILEN:AUTO ON  
 Selects the most sensible parameters automatically.

**Manual operation:** See ["Impulse Length"](#) on page 92

**[:SOURce<hw>]:BB:BT0oth:FILTer:OSAMpling <OSampling>**

Sets the upsampling factor.

**Parameters:**

<OSampling> integer  
 Range: 1 to 32  
 \*RST: 10

**Example:** SOURce1:BB:BT0:FILT:OSAM 10  
 Sets the upsampling factor to 10.

**Manual operation:** See ["Oversampling"](#) on page 92

**[:SOURce<hw>]:BB:BT0oth:FILTer:OSAMpling:AUTO[:STATE] <State>**

Activates the upsampling factor state. If activated, the most sensible parameter values are selected. The value depends on the coherence check. If deactivated, the values can be changed manually.

**Parameters:**

<State> 1 | ON | 0 | OFF  
 \*RST: 1

**Example:** SOURce1:BB:BT0:FILT:OSAM:AUTO ON  
 The most sensible parameters are selected automatically.

**Manual operation:** See ["Oversampling"](#) on page 92

**[:SOURce<hw>]:BB:BT0oth:FILTer:MINDeX <MIndex>**

Queries the modulation index resulting from the entered frequency deviation value.

**Parameters:**

<MIndex> string

**Example:** SOURce1:BB:BT0:FILT:MINDeX?  
 Queries the modulation index  
 // Response: "0.5"

**Manual operation:** See ["Modulation Index"](#) on page 94

**[:SOURce<hw>]:BB:BT0oth:FILTer:MTYPE <MType>**

Queries the modulation type used for the current packet selection.

**Parameters:**

<MType> string

**Example:** SOURce1:BB:BT0:FILT:MTYPE?  
 Queries the modulation type

**Manual operation:** See ["Modulation Type"](#) on page 93

---

**[[:SOURce<hw>]:BB:BT0oth:MSETtings:FDEVIation <FDeviation>**

Sets the frequency deviation.

**Parameters:**

<FDeviation>	float
Range:	Depends on Bluetooth mode
Increment:	0.1
*RST:	depends on Bluetooth mode

**Example:** SOURce1:BB:BT0:MSET:FDEV 160  
Sets a frequency deviation of 160 kHz.

**Manual operation:** See ["Frequency Deviation"](#) on page 93

---

**[[:SOURce<hw>]:BB:BT0oth:FILTer:PARAmeter:APCO25 <Apco25>**

Sets the roll-off factor for filter type APCO25.

**Parameters:**

<Apco25>	float
Range:	0.05 to 0.99
Increment:	0.01
*RST:	0.2

**Example:** SOURce1:BB:BT0:FILT:PAR:APCO25 0.2  
Sets the roll-off factor to 0.2 for filter type APCO25.

**Manual operation:** See ["Roll Off Factor / BxT"](#) on page 91  
See ["Roll Off Factor / BxT"](#) on page 92

---

**[[:SOURce<hw>]:BB:BT0oth:FILTer:PARAmeter:COSSine <Cosine>**

Sets the roll-off factor for the Cosine filter type.

**Parameters:**

<Cosine>	float
Range:	0 to 1
Increment:	0.01
*RST:	0.1

**Example:** SOURce1:BB:BT0:FILT:PAR:COS 0.35  
Sets the roll-off factor to 0.35 for filter type Cosine.

**Manual operation:** See ["Roll Off Factor / BxT"](#) on page 91  
See ["Roll Off Factor / BxT"](#) on page 92

---

**[[:SOURce<hw>]:BB:BT0oth:FILTer:PARAmeter:FGAUSS <FGauss>**

Sets the B x T for the Gauss filter type.

**Parameters:**

<FGauss> float  
 Range: 0.15 to 2.5  
 Increment: 0.01  
 \*RST: 0.5

**Example:**

SOURce1:BB:BTO:FILT:PAR:FGA 0.5  
 Sets B x T to 0.5 for the Gauss filter type for the GFSK section of the packet.

**Manual operation:**

See "Roll Off Factor / BxT" on page 91  
 See "Roll Off Factor / BxT" on page 92

**[:SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:GAUSs <Gauss>**

Sets the B x T for the Gauss filter type.

**Parameters:**

<Gauss> float  
 Range: 0.15 to 2.5  
 Increment: 0.01  
 \*RST: 0.5

**Example:**

SOURce1:BB:BTO:FILT:PAR:GAUS 0.5  
 Sets B x T to 0.5 for the Gauss filter type for  $\pi/4$  DQPSK or 8DPSK sections.

**Manual operation:**

See "Roll Off Factor / BxT" on page 91  
 See "Roll Off Factor / BxT" on page 92

**[:SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:LPASs <LPass>**

Sets the cut off frequency factor for a lowpass filter (ACP Opt.).

**Parameters:**

<LPass> float  
 Range: 0.05 to 2  
 Increment: 0.01  
 \*RST: 0.5

**Example:**

SOURce1:BB:BTO:FILT:PAR:LPAS 1  
 Sets the cut off frequency factor for a lowpass filter.

**Manual operation:**

See "Roll Off Factor / BxT" on page 91  
 See "Roll Off Factor / BxT" on page 92  
 See "Cut Off Frequency Factor" on page 93

**[:SOURce<hw>]:BB:BTOoth:FILTer:PARAmeter:PGAuss <PGauss>**

Sets the B x T for the Pure Gauss filter type.

**Parameters:**

<PGauss> float  
 Range: 0.15 to 2.5  
 Increment: 0.01  
 \*RST: 0.5

**Example:**

SOURce1:BB:BTO:FILT:PAR:PGA 0.5  
 Sets B x T to 0.5 for the Pure Gauss filter type.

**Manual operation:**

See ["Roll Off Factor / BxT"](#) on page 91  
 See ["Roll Off Factor / BxT"](#) on page 92

**[:SOURce<hw>]:BB:BTOoth:FILT:PAR:RCOSine <RCosine>**

Sets the roll-off factor for the Root Cosine filter type.

**Parameters:**

<RCosine> float  
 Range: 0 to 1  
 Increment: 0.01  
 \*RST: 0.4

**Example:**

SOURce1:BB:BTO:FILT:PAR:RCOS 0.22  
 Sets the roll-off factor to 0.22 for filter type Root Cosine.

**Manual operation:**

See ["Roll Off Factor / BxT"](#) on page 91  
 See ["Roll Off Factor / BxT"](#) on page 92

**[:SOURce<hw>]:BB:BTOoth:FILT:PAR:SPHase <SPhase>**

Sets the BxT for the split phase filter type.

**Parameters:**

<SPhase> float  
 Range: 0.15 to 2.5  
 Increment: 0.01  
 \*RST: 2

**Example:**

SOURce1:BB:BTO:FILT:PAR:SPH 0.5  
 Sets B x T to 0.5 for the Split Phase filter type.

**Manual operation:**

See ["Roll Off Factor / BxT"](#) on page 91  
 See ["Roll Off Factor / BxT"](#) on page 92

**[:SOURce<hw>]:BB:BTOoth:SRATe:VARiation <Variation>**

Sets the symbol rate.

**Parameters:**

<Variation> float  
 Range: 4E2 to 15E6  
 Increment: 1E-3  
 \*RST: 1E6

**Example:**

BB:BTO:SRAT:VAR 1  
 sets the symbol rate variation to 1 MHz.

**Manual operation:** See "[Symbol Rate Variation](#)" on page 94

## 5.10 Power ramping commands

<a href="#">[:SOURce&lt;hw&gt;]:BB:BTOoth:PRAMping:FOFFset</a> .....	192
<a href="#">[:SOURce&lt;hw&gt;]:BB:BTOoth:PRAMping:RFUNction</a> .....	192
<a href="#">[:SOURce&lt;hw&gt;]:BB:BTOoth:PRAMping:ROFFset</a> .....	193
<a href="#">[:SOURce&lt;hw&gt;]:BB:BTOoth:PRAMping:RTIME</a> .....	193

---

### **[[:SOURce<hw>]:BB:BTOoth:PRAMping:FOFFset <FOffset>**

Sets the offset of the falling edge of a burst. The offset is specified by the selected number of symbols.

Negative values shift the falling edge to earlier positions, which results in a corresponding number of skipped symbols at the end of the burst.

Positive values shift the falling edge to later positions, which results in a corresponding number of added 0 padding symbols following the burst.

**Parameters:**

<FOffset> integer  
 Range: -32 to 32  
 Increment: 1  
 \*RST: 0

**Example:**

BB:BTO:PRAM:FOFF 8  
 Adds eight symbols at the end of the burst.

**Manual operation:** See "[Fall Offset](#)" on page 97

---

### **[[:SOURce<hw>]:BB:BTOoth:PRAMping:RFUNction <RFunction>**

The command selects the form of the transmitted power, i.e. the shape of the rising and falling edges during power ramp control.

**Parameters:**

<RFunction> LINear | COSine  
 \*RST: COSine



**Example:** `BB:BTO:PRAM:RFUN LIN`  
sets linear shape for the rising and falling edges during power ramp control.

**Manual operation:** See "[Ramp Function](#)" on page 96

**[:SOURce<hw>]:BB:BTOoth:PRAMping:ROFFset <ROffset>**

Sets the offset of the rising edge of a burst. The offset is specified by the selected number of symbols.

Negative values shift the rising edge to earlier positions, which results in a corresponding number of added 0 padding symbols before the burst.

Positive values shift the rising edge to later positions, which results in a corresponding number of skipped symbols at the beginning of the burst.

**Parameters:**

<ROffset> integer  
Range: -32 to 32  
Increment: 1  
\*RST: -2

**Example:** `BB:BTO:PRAM:ROFF 8`  
Skips eight symbols at the beginning of the burst.

**Manual operation:** See "[Rise Offset](#)" on page 96

**[:SOURce<hw>]:BB:BTOoth:PRAMping:RTIME <RTIME>**

Sets the ramp time, which extends the burst by a corresponding number of 0 padding symbols at the beginning and the end of a burst. During this period of time, power ramping is based on the specified ramp function.

**Parameters:**

<RTIME> integer  
Range: 1 to 32  
Increment: 1  
\*RST: 2

**Example:** `BB:BTO:PRAM:TIME 2`  
Extends the burst by 2 symbols at the beginning and end of the burst.

**Manual operation:** See "[Ramp Time](#)" on page 96

## 5.11 Trigger commands

### Example: To configure an external trigger

```
// *****
// Configure trigger in armed retrigger mode, set source, enable
// synchronization to external trigger, set external
// inhibit and delay.
// *****
SOURCEl:BB:BT0oth:TRIGger:SEQUence ARET
SOURCEl:BB:BT0oth:TRIGger:SOURce EGT1
SOURCEl:BB:BT0oth:TRIGger:EXTernal:SYNChronize:OUTPut 1
SOURCEl:BB:BT0oth:TRIGger:EXTernal:INHibit 100
SOURCEl:BB:BT0oth:TRIGger:EXTernal:DELAy 10
```

### Example: To configure an internal trigger

```
// *****
// Configure trigger in single mode.
// Set source to internal, specify signal duration
// unit and duration.
// *****
SOURCEl:BB:BT0oth:TRIGger:SEQUence SING
SOURCEl:BB:BT0oth:TRIGger:SOURce INT
SOURCEl:BB:BT0oth:TRIGger:SLUNit SEQ

SOURCEl:BB:BT0oth:TRIGger:SLUNit FRAME
SOURCEl:BB:BT0oth:TRIGger:SLENGth 2

// *****
// Alternatively configure trigger in armed retrigger mode.
// Set source to internal. Enable Bluetooth signal, start
// the trigger - signal generation starts.
// Stop signal generation and wait for a trigger
// event to restart signal generation.
// Query the current trigger signal generation status.
// *****
SOURCEl:BB:BT0oth:TRIGger:SEQUence ARETrigger
SOURCEl:BB:BT0oth:TRIGger:SOURce INT
SOURCEl:BB:BT0oth:STATe 1
SOURCEl:BB:BT0oth:TRIGger:EXEcute
```

<a href="#">[:SOURCE&lt;hw&gt;]:BB:BT0oth[:TRIGger]:SEQUence.....</a>	195
<a href="#">[:SOURCE&lt;hw&gt;]:BB:BT0oth:TRIGger:SOURce.....</a>	195
<a href="#">[:SOURCE&lt;hw&gt;]:BB:BT0oth:TRIGger:RMODE.....</a>	196
<a href="#">[:SOURCE&lt;hw&gt;]:BB:BT0oth:TRIGger:TIME:DATE.....</a>	196
<a href="#">[:SOURCE&lt;hw&gt;]:BB:BT0oth:TRIGger:TIME:TIME.....</a>	196
<a href="#">[:SOURCE&lt;hw&gt;]:BB:BT0oth:TRIGger:TIME[:STATe].....</a>	197
<a href="#">[:SOURCE&lt;hw&gt;]:BB:BT0oth:TRIGger:SLENGth.....</a>	197
<a href="#">[:SOURCE&lt;hw&gt;]:BB:BT0oth:TRIGger:SLUNit.....</a>	197

<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:TRIGger:ARM:EXECute</code> .....	198
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:TRIGger:EXECute</code> .....	198
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:TRIGger[:EXTeRnal]:SYNChronize:OUTPut</code> .....	198
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:TRIGger[:EXTeRnal]:DELay</code> .....	198
<code>[:SOURce&lt;hw&gt;]:BB:BT0oth:TRIGger[:EXTeRnal]:INHibit</code> .....	199

---

### `[:SOURce<hw>]:BB:BT0oth[:TRIGger]:SEQUence <Sequence>`

Selects the trigger mode:

- `AUTO` = auto
- `RETRigger` = retrigger
- `AAUTo` = armed auto
- `ARETRigger` = armed retrigger
- `SINGLE` = single

#### Parameters:

`<Sequence>`            `AUTO` | `RETRigger` | `AAUTo` | `ARETRigger` | `SINGLE`  
 \*RST:            `AUTO`

**Example:**            See [Example "To configure an internal trigger"](#) on page 194.

**Manual operation:** See ["Mode"](#) on page 98

---

### `[:SOURce<hw>]:BB:BT0oth:TRIGger:SOURce <Source>`

Selects the trigger signal source and determines the way the triggering is executed. Provided are:

- Internal triggering by a command (`INTernal`)
- External trigger signal via one of the local or global connectors
  - `EGT1` | `EGT2`: External global trigger
  - `EGC1` | `EGC2`: External global clock
- In primary-secondary instrument mode, the external baseband synchronization signal (`BBSY`)
- `OBASeband` | `BEXTeRnal` | `EXTeRnal`: Setting only  
 Provided only for backward compatibility with other Rohde & Schwarz signal generators.  
 The R&S SMM100A accepts these values and maps them automatically as follows:  
`EXTeRnal` = `EGT1`, `BEXTeRnal` = `EGT2`, `OBASeband` = `INTA`

#### Parameters:

`<Source>`            `INTernal` | `EGT1` | `EGT2` | `EGC1` | `EGC2` | `EXTeRnal` | `BBSY`  
 \*RST:            `INTernal`

**Example:**            See [Example "To configure an external trigger"](#) on page 194.

**Example:**            See [Example "To configure an internal trigger"](#) on page 194.

**Manual operation:** See ["Source"](#) on page 100

---

**[ :SOURce<hw> ]:BB:BT0oth:TRIGger:RMODE <RMode>**

Queries signal generation status for all trigger modes with Bluetooth modulation on.

**Parameters:**

<RMode>                    RUN | STOP  
                              \*RST:        STOP

**Example:**                See [Example "To configure an internal trigger"](#) on page 194.

**Manual operation:**    See ["Running/Stopped"](#) on page 99

---

**[ :SOURce<hw> ]:BB:BT0oth:TRIGger:TIME:DATE <Year>, <Month>, <Day>**

Sets the date for a time-based trigger signal. For trigger modes single or armed auto, you can activate triggering at this date via the following command:

SOURce<hw>:BB:<DigStd>:TRIGger:TIME:STATe

<DigStd> is the mnemonic for the digital standard, for example, ARB. Time-based triggering behaves analogously for all digital standards that support this feature.

**Parameters:**

<Year>                    integer  
                              Range:        1980 to 9999  
  
 <Month>                  integer  
                              Range:        1 to 12  
  
 <Day>                    Range:        1 to 31  
                              Increment:    1

**Example:**                See example "Configure a time-based trigger signal" in the sub-chapter "Trigger Commands" of the chapter "SOURce:BB:ARB subsystem" in the R&S SMM100A user manual.

**Manual operation:**    See ["Trigger Time"](#) on page 99

---

**[ :SOURce<hw> ]:BB:BT0oth:TRIGger:TIME:TIME <Time>**

Sets the time for a time-based trigger signal. For trigger modes single or armed auto, you can activate triggering at this time via the following command:

SOURce<hw>:BB:<DigStd>:TRIGger:TIME:STATe

<DigStd> is the mnemonic for the digital standard, for example, ARB. Time-based triggering behaves analogously for all digital standards that support this feature.

**Parameters:**

<Time>                    string

**Example:**                See example "Configure a time-based trigger signal" in the sub-chapter "Trigger Commands" of the chapter "SOURce:BB:ARB subsystem" in the R&S SMM100A user manual.

**Manual operation:**    See ["Trigger Time"](#) on page 99

---

**[[:SOURce<hw>]:BB:BT0oth:TRIGger:TIME[:STATe] <Hour>, <Minute>, <Second>**

Activates time-based triggering with a fixed time reference. If activated, the R&S SMM100A triggers signal generation when its operating system time matches a specified time.

Specify the trigger date and trigger time with the following commands:

```
SOURce<hw>:BB:<DigStd>:TRIGger:TIME:DATE
```

```
SOURce<hw>:BB:<DigStd>:TRIGger:TIME:TIME
```

<DigStd> is the mnemonic for the digital standard, for example, ARB. Time-based triggering behaves analogously for all digital standards that support this feature.

**Parameters:**

<Hour>	integer	
	Range:	0 to 23
<Minute>	integer	
	Range:	0 to 59
<Second>	integer	
	Range:	0 to 59

**Example:** See example "Configure a time-based trigger signal" in the sub-chapter "Trigger Commands" of the chapter "SOURce:BB:ARB subsystem" in the R&S SMM100A user manual.

**Manual operation:** See ["Time Based Trigger"](#) on page 99

---

**[[:SOURce<hw>]:BB:BT0oth:TRIGger:SLENgth <SLength>**

Defines the length of the signal sequence that is output in the SINGLE trigger mode.

**Parameters:**

<SLength>	integer	
	Range:	1 to 7000
	*RST:	1

**Example:** See [Example "To configure an internal trigger"](#) on page 194.

**Manual operation:** See ["Signal Duration"](#) on page 99

---

**[[:SOURce<hw>]:BB:BT0oth:TRIGger:SLUNit <SIUnit>**

Defines the unit for the entry of the signal sequence length.

**Parameters:**

<SIUnit>	FRAMe   SEQUence   EVENt
----------	--------------------------

**FRAMe**

A single frame is generated after a trigger event.

**SEQUence**

A single sequence is generated after a trigger event.

\*RST: SEQuence

**Example:** See [Example "To configure an internal trigger"](#) on page 194.

**Manual operation:** See ["Signal Duration Unit"](#) on page 99

#### **[ :SOURce<hw> ]:BB:BT0oth:TRIGger:ARM:EXECute**

Stops signal generation; a subsequent trigger event restarts signal generation.

**Example:** See [Example "To configure an internal trigger"](#) on page 194.

**Usage:** Event

**Manual operation:** See ["Arm"](#) on page 100

#### **[ :SOURce<hw> ]:BB:BT0oth:TRIGger:EXECute**

Executes a trigger.

**Example:** See [Example "To configure an internal trigger"](#) on page 194.

**Usage:** Event

**Manual operation:** See ["Execute Trigger"](#) on page 100

#### **[ :SOURce<hw> ]:BB:BT0oth:TRIGger[:EXTernal]:SYNChronize:OUTPut <Output>**

Enables signal output synchronous to the trigger event.

**Parameters:**

<Output> 1 | ON | 0 | OFF

\*RST: 1

**Example:** See [Example "To configure an external trigger"](#) on page 194.

**Manual operation:** See ["Sync. Output to Ext. Trigger/Sync. Output to Trigger"](#) on page 100

#### **[ :SOURce<hw> ]:BB:BT0oth:TRIGger[:EXTernal]:DELay <Delay>**

Sets the trigger delay.

**Parameters:**

<Delay> float

Range: 0 to 2147483647

Increment: 0.01

\*RST: 0

**Example:** See [Example "To configure an external trigger"](#) on page 194.

**Manual operation:** See ["External Delay/Trigger Delay"](#) on page 101

**[[:SOURce<hw>]:BB:BT0oth:TRIGger[:EXTernal]:INHibit <Inhibit>**

Specifies the number of samples by which a restart is to be inhibited following an external trigger event.

**Parameters:**

<Inhibit>                    integer  
                                  Range:     0 to 21.47\*symbRate  
                                  \*RST:     0

**Example:**                    See [Example "To configure an external trigger"](#) on page 194.

**Manual operation:**        See ["External Inhibit/Trigger Inhibit"](#) on page 101

## 5.12 Marker commands

**Example: To configure a marker signal**

```
// *****
// Configure marker mode: set a marker at ARB sequence start.
// *****
SOURcel:BB:BT0oth:TRIGger:OUTPut1:MODE REStart
// SOURcel:BB:BT0oth:TRIGger:OUTPut1:MODE StARt
// SOURcel:BB:BT0oth:TRIGger:OUTPut1:MODE ACTive

// *****
// Alternatively configure pulse marker. Set pulse
// divider and frequency
// *****
SOURcel:BB:BT0oth:TRIGger:OUTPut1:MODE PULSe
SOURcel:BB:BT0oth:TRIGger:OUTPut1:PULSe:DIVider 2
SOURcel:BB:BT0oth:TRIGger:OUTPut1:PULSe:FREquency?
// 500000

// *****
// Alternatively configure bit pattern marker. Specify pattern.
// *****
SOURcel:BB:BT0oth:TRIGger:OUTPut1:MODE PATtern
SOURcel:BB:BT0oth:TRIGger:OUTPut1:PATtern #H2,2
// *****
// Alternatively configure on/off ratio marker. Set on/off time.
// *****
SOURcel:BB:BT0oth:TRIGger:OUTPut1:MODE RAT
SOURcel:BB:BT0oth:TRIGger:OUTPut1:ONtime 40000
SOURcel:BB:BT0oth:TRIGger:OUTPut1:OFFtime 20000
```

**Example: To configure marker delay**

```
// *****
// Enable fixed marker delay. Query the limit for minimum and maximum
// marker delay. Set delay for the marker signal output.
// *****
SOURce1:BB:BTOoth:TRIGger:OUTPut1:DElay 1600
```

[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:DElay.....	200
[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:FESHift.....	200
[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:MODE.....	201
[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:ONTime.....	201
[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:OFFTime.....	201
[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:PATtern.....	202
[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:PINdex.....	202
[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:PULSe:DIVider.....	202
[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:PULSe:FREQuency?.....	202
[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:RESHift.....	203

**[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:DElay <Delay>**

Defines the delay between the signal on the marker outputs and the start of the signals.

**Parameters:**

<Delay>	float
	Range: 0 to 16777215
	Increment: 1
	*RST: 0

**Example:** See [Example "To configure marker delay"](#) on page 200.

**Manual operation:** See ["Delay"](#) on page 104

**[SOURce<hw>]:BB:BTOoth:TRIGger:OUTPut<ch>:FESHift <Shift>**

Shifts the falling edge of the marker the specified number of samples.

Negative values result in a shift back of the marker edge.

**Parameters:**

<Shift>	float
	Range: dynamic to dynamic
	Increment: 1E-6
	*RST: 0

**Example:** SOURce1:BB:BTO:TRIG:OUTP1:FESH 10  
Shifts the falling edge of marker 1 by 75 samples.

**Example:** See also [Example "To configure a marker signal"](#) on page 199.

**Manual operation:** See ["Mode"](#) on page 102



---

```
[ :SOURce<hw>]:BB:BT0oth:TRIGger:OUTPut<ch>:MODE <Mode>
```

Defines the signal for the selected marker output.

**Parameters:**

<Mode>

REStart | START | ACTive | PULSe | PATtern | RATio | IACTive

**REStart**

A marker signal is generated at the start of each signal sequence.

**START**

A marker signal is generated at the start of each event/frame.

**ACTive**

The marker masks the active part of the event/frame. At the start of each burst, the marker signal changes to high. It changes back to low after the end of each burst.

**PULSe**

A regular marker signal is generated. The clock frequency is defined by entering a divider. The frequency is derived by dividing the symbol rate by the divider. The input box for divider opens when "Pulse" is selected, and the resulting pulse frequency is displayed below.

**PATtern**

A marker signal that is defined by a bit pattern is generated. The pattern has a maximum length of 32 bits and is defined in an input field which opens when pattern is selected.

**RATio**

A regular marker signal corresponding to the "Time Off" / "Time On" specifications in the commands

`SOURce1:BB:BT0:TRIGger:OUTPut:OFFTime` and

`SOURce1:BB:BT0:TRIGger:OUTPut:ONTime` is generated.

**IACTive**

The marker masks the inactive part of the event/frame. At the start of each burst, the marker signal changes to low. It changes back to high after the end of each burst.

\*RST: REStart

**Example:** See [Example "To configure a marker signal"](#) on page 199.

**Manual operation:** See ["Mode"](#) on page 102

---

```
[ :SOURce<hw>]:BB:BT0oth:TRIGger:OUTPut<ch>:ONTime <OnTime>
```

```
[ :SOURce<hw>]:BB:BT0oth:TRIGger:OUTPut<ch>:OFFTime <OffTime>
```

Sets the duration of the ON and OFF periods.

**Parameters:**

<OffTime>

integer

Range: 1 to 16777215

\*RST: 1

**Example:** See [Example "To configure a marker signal"](#) on page 199.

**Manual operation:** See ["Mode"](#) on page 102

**[:SOURce<hw>]:BB:BT0oth:TRIGger:OUTPut<ch>:PATTern <Pattern>, <BitCount>**

Selects the data for a pattern.

**Parameters:**

<Pattern>	numeric
	*RST: #H2
<BitCount>	integer
	Range: 1 to 64
	*RST: 2

**Example:** See [Example "To configure a marker signal"](#) on page 199.

**Manual operation:** See ["Mode"](#) on page 102

**[:SOURce<hw>]:BB:BT0oth:TRIGger:OUTPut<ch>:PINdex <PIndex>**

For Bluetooth LE data packets higher than one, sets the packet index. The index corresponds to the transmitted Tx event during the connection interval.

**Parameters:**

<PIndex>	integer
	Range: 1 to depends on "No. Of Tx Packets/Event" parameter
	*RST: 1

**Manual operation:** See ["Mode"](#) on page 102

**[:SOURce<hw>]:BB:BT0oth:TRIGger:OUTPut<ch>:PULSe:DIVider <Divider>**

Sets the divider for the clock frequency.

**Parameters:**

<Divider>	integer
	Range: 2 to 1024
	*RST: 2

**Example:** See [Example "To configure a marker signal"](#) on page 199.

**Manual operation:** See ["Mode"](#) on page 102

**[:SOURce<hw>]:BB:BT0oth:TRIGger:OUTPut<ch>:PULSe:FREQUency?**

Queries the marker pulse frequency.

**Return values:**

<Frequency> float  
 Range: 2 to 1024  
 Increment: 1E-3  
 \*RST: 2

**Example:** See [Example "To configure a marker signal"](#) on page 199.

**Usage:** Query only

**Manual operation:** See ["Mode"](#) on page 102

**[:SOURce<hw>]:BB:BT0oth:TRIGger:OUTPut<ch>:RESHift <Shift>**

Shifts the rising edge of the marker the specified number of samples.

Negative values result in a shift back of the marker edge.

**Parameters:**

<Shift> float  
 Range: dynamic to dynamic  
 Increment: 1E-6  
 \*RST: 0

**Example:** SOURce1:BB:BT0:TRIG:OUTPut2:RESH -20  
 Shifts back the rising edge of marker 2 by 20 samples.

**Example:** See also [Example "To configure a marker signal"](#) on page 199.

**Manual operation:** See ["Mode"](#) on page 102

## 5.13 Clock commands

This section lists the remote control commands, necessary to configure the clock.

**Example: To configure clock settings**

```
// *****
// Select internal clock.
// *****
SOURce1:BB:BT0oth:CLOCK:SOURce INTernal
// *****
// Alternatively select external clock. Set its mode.
// *****
SOURce1:BB:BT0oth:CLOCK:MODE SAMP
```

[\[:SOURce<hw>\]:BB:BT0oth:CLOCK:SOURce.....](#) 204

---

**[ :SOURce<hw> ] :BB:BT0oth:CLOCK:SOURce <Source>**

Selects the clock source:

- **INTernal**: Internal clock reference
- **ELCLock**: External local clock
- **EXTernal = ELCLock**: Setting only  
Provided for backward compatibility with other Rohde & Schwarz signal generators

**Parameters:**

<Source>            INTernal  
                      \*RST:        INTernal

**Example:**            See [Example "To configure clock settings"](#) on page 203.

**Manual operation:** See ["Clock Source"](#) on page 105

# Glossary: Specifications

## Symbols

[1]: Bluetooth SIG: Bluetooth Core Specification v5.4

<https://www.bluetooth.com/specifications/bluetooth-core-specification>

## List of commands

[:SOURce<hw>]:BB:BT00th:BCRole.....	130
[:SOURce<hw>]:BB:BT00th:BCText?.....	108
[:SOURce<hw>]:BB:BT00th:BMODE.....	109
[:SOURce<hw>]:BB:BT00th:CCRC:STATe.....	130
[:SOURce<hw>]:BB:BT00th:CLIPping:LEVel.....	186
[:SOURce<hw>]:BB:BT00th:CLIPping:MODE.....	186
[:SOURce<hw>]:BB:BT00th:CLIPping:STATe.....	186
[:SOURce<hw>]:BB:BT00th:CLOCK:SOURce.....	204
[:SOURce<hw>]:BB:BT00th:CTYPe.....	131
[:SOURce<hw>]:BB:BT00th:DCYCLE.....	131
[:SOURce<hw>]:BB:BT00th:DTTTest:DTTState.....	113
[:SOURce<hw>]:BB:BT00th:DTTTest:FDDeviation.....	113
[:SOURce<hw>]:BB:BT00th:DTTTest:FDRate.....	114
[:SOURce<hw>]:BB:BT00th:DTTTest:MIMode.....	114
[:SOURce<hw>]:BB:BT00th:DTTTest:NPPSet.....	114
[:SOURce<hw>]:BB:BT00th:DTTTest:SPHase.....	115
[:SOURce<hw>]:BB:BT00th:DTTTest:STDefault.....	115
[:SOURce<hw>]:BB:BT00th:DTTTest:TABLE.....	115
[:SOURce<hw>]:BB:BT00th:DTTTest:TABLE:LONG:SET<ch>:CFOffset.....	115
[:SOURce<hw>]:BB:BT00th:DTTTest:TABLE:LONG:SET<ch>:MINDex.....	116
[:SOURce<hw>]:BB:BT00th:DTTTest:TABLE:LONG:SET<ch>:STATe.....	117
[:SOURce<hw>]:BB:BT00th:DTTTest:TABLE:LONG:SET<ch>:STERror.....	117
[:SOURce<hw>]:BB:BT00th:DTTTest:TABLE:SHORT:SET<ch>:CFOffset.....	117
[:SOURce<hw>]:BB:BT00th:DTTTest:TABLE:SHORT:SET<ch>:STATe.....	118
[:SOURce<hw>]:BB:BT00th:DTTTest:TABLE:SHORT:SET<ch>:STERror.....	118
[:SOURce<hw>]:BB:BT00th:DTTTest:TPConfiguration:TPINterval.....	184
[:SOURce<hw>]:BB:BT00th:DTTTest:TPConfiguration:UPLength.....	184
[:SOURce<hw>]:BB:BT00th:DTTTest:TPConfiguration:UPSource.....	185
[:SOURce<hw>]:BB:BT00th:DURATION.....	134
[:SOURce<hw>]:BB:BT00th:ECONfig:PCONfig:ANTGain<ch0>.....	154
[:SOURce<hw>]:BB:BT00th:ECONfig:PCONfig:ANTNumber.....	154
[:SOURce<hw>]:BB:BT00th:ECONfiguration:ACTable:CHANnel<ch0>:STATe.....	136
[:SOURce<hw>]:BB:BT00th:ECONfiguration:ADINterval.....	137
[:SOURce<hw>]:BB:BT00th:ECONfiguration:AEDelay.....	137
[:SOURce<hw>]:BB:BT00th:ECONfiguration:AEINterval.....	137
[:SOURce<hw>]:BB:BT00th:ECONfiguration:APINterval.....	138
[:SOURce<hw>]:BB:BT00th:ECONfiguration:DCTable:CHANnel<ch0>:STATe.....	136
[:SOURce<hw>]:BB:BT00th:ECONfiguration:LCMode.....	138
[:SOURce<hw>]:BB:BT00th:ECONfiguration:LTKey.....	139
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:AADDRESS.....	151
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ACAD.....	156
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ACAD:APATtern.....	157
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ACAD:ASELectioN.....	157
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ACASSigned.....	158
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ACID.....	158
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ADID.....	158
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:ALAP.....	159

[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:ALEngth.....	159
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:AMODE.....	160
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:ANUap.....	160
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:AOFFset.....	161
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:AOUNits.....	161
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:APHY.....	161
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:ASID.....	162
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:ATYPE.....	180
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CACCuracy.....	162
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CECount.....	181
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CID.....	162
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CINStart.....	163
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CINterval.....	163
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CIValue.....	151
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CPResent.....	152
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CSElection.....	152
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CTIME.....	153
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CTReq.....	153
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:CTYPE.....	153
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:DATA.....	163
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:DATA:DPATern.....	164
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:DATA:DSElection.....	164
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:DCMTable:CHANnel<ch0>:STATe.....	136
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:DLEngth.....	165
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:DWHitening.....	150
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:ECODE.....	165
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:ECOUNTER.....	165
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:EDIVersifier.....	166
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:EHEader:STATe.....	166
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:EHFLags:AADDRESS:STATe.....	166
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:EHFLags:ADINfo:STATe.....	167
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:EHFLags:APTR:STATe.....	167
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:EHFLags:CINfo:STATe.....	167
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:EHFLags:SINfo:STATe.....	167
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:EHFLags:TADDRESS:STATe.....	168
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:EHFLags:TPOWER:STATe.....	168
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:FSBit<ch0>:STATe.....	168
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:FSLength.....	169
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:HLEngth.....	169
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:ICASSigned.....	158
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:ICID.....	158
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:ID.....	181
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:ILAP.....	159
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:INUap.....	160
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:LCTimeout.....	169
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:LPECounter.....	181
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:MCLReq.....	181
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:MIVector.....	170
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:MNINterval.....	139
[:SOURce<hw>]:BB:BT0oth:ECOnfiguration:PCONfiguration:MROctets.....	170

[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MRTIME.....	171
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MSKD.....	170
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MTOCtets.....	170
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MTSPHy:L1M:STATe.....	172
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MTSPHy:L2M:STATe.....	172
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MTSPHy:LCOD:STATe.....	172
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MTTime.....	171
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MUCHannels.....	171
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:MXINterval.....	140
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NCINterval.....	172
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NLCTimeout.....	173
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NSLatency.....	173
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NSValue.....	152
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NWOFset.....	173
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:NWSize.....	174
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:OADJust.....	174
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:OFFSet<ch0>:STATe.....	140
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:OFFSet<ch0>:VALue.....	140
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PAINterval.....	141
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PHY:L1M:STATe.....	182
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PHY:L2M:STATe.....	182
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PHY:LCOD:STATe.....	182
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PHYS:L1M:STATe.....	174
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PHYS:L2M:STATe.....	174
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PHYS:LCOD:STATe.....	174
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:PPERiodicity.....	141
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RAType.....	175
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RCECount.....	142
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:ROPCode.....	175
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RPHYs:L1M:STATe.....	176
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RPHYs:L2M:STATe.....	176
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RPHYs:LCOD:STATe.....	176
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:RVEctor.....	176
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SCACcuracy.....	177
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SCASsigned.....	158
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SCECounter.....	183
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SCID.....	158
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SID.....	183
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SIVector.....	170
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SLAP.....	159
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SLATency.....	177
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SNUap.....	160
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SOUNits.....	177
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SPBit.....	184
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SPOFset.....	178
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SSKD.....	170
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:SSValue.....	152
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:STMPHy:L1M:STATe.....	172
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:STMPHy:L2M:STATe.....	172
[:SOURce<hw>]:BB:BT0oth:ECONfiguration:PCONfiguration:STMPHy:LCOD:STATe.....	172



[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:SVNumber.....	178
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:TAType.....	175
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:TLAP.....	159
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:TNUap.....	160
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:TPHYs:L1M:STATe.....	176
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:TPHYs:L2M:STATe.....	176
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:TPHYs:LCOD:STATe.....	176
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:TPOWer.....	178
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:UTYPE.....	179
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:VNUmber.....	179
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:WOffset.....	180
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PCONfiguration:WSize.....	180
[:SOURce<hw>]:BB:BT00th:ECONfiguration:PNUmber.....	142
[:SOURce<hw>]:BB:BT00th:ECONfiguration:SDCI?.....	142
[:SOURce<hw>]:BB:BT00th:ECONfiguration:SINTerval.....	143
[:SOURce<hw>]:BB:BT00th:ECONfiguration:SWINdow.....	143
[:SOURce<hw>]:BB:BT00th:ECONfiguration:Woinfo?.....	143
[:SOURce<hw>]:BB:BT00th:ECONfiguration:WSInfo?.....	144
[:SOURce<hw>]:BB:BT00th:FILTer:ILENght.....	187
[:SOURce<hw>]:BB:BT00th:FILTer:ILENght:AUTO[STATe].....	187
[:SOURce<hw>]:BB:BT00th:FILTer:MINdex.....	188
[:SOURce<hw>]:BB:BT00th:FILTer:MTYPE.....	188
[:SOURce<hw>]:BB:BT00th:FILTer:OSAMpling.....	188
[:SOURce<hw>]:BB:BT00th:FILTer:OSAMpling:AUTO[STATe].....	188
[:SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:APCO25.....	189
[:SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:COsine.....	189
[:SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:FGAuss.....	189
[:SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:GAUSSs.....	190
[:SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:LPASSs.....	190
[:SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:PGAuss.....	190
[:SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:RCOSine.....	191
[:SOURce<hw>]:BB:BT00th:FILTer:PARAmeter:SPHase.....	191
[:SOURce<hw>]:BB:BT00th:FILTer:TYPE.....	187
[:SOURce<hw>]:BB:BT00th:MFORmat.....	133
[:SOURce<hw>]:BB:BT00th:MSETtings:FDEVIation.....	189
[:SOURce<hw>]:BB:BT00th:PCONfiguration:ACKNowledgeMent.....	120
[:SOURce<hw>]:BB:BT00th:PCONfiguration:BDALap.....	121
[:SOURce<hw>]:BB:BT00th:PCONfiguration:BDANap.....	121
[:SOURce<hw>]:BB:BT00th:PCONfiguration:BDAUap.....	122
[:SOURce<hw>]:BB:BT00th:PCONfiguration:CODevice.....	122
[:SOURce<hw>]:BB:BT00th:PCONfiguration:DATA.....	122
[:SOURce<hw>]:BB:BT00th:PCONfiguration:DATA:DPATtern.....	123
[:SOURce<hw>]:BB:BT00th:PCONfiguration:DATA:DSELection.....	123
[:SOURce<hw>]:BB:BT00th:PCONfiguration:DATA:VDPAattern.....	123
[:SOURce<hw>]:BB:BT00th:PCONfiguration:DATA:VDSElection.....	124
[:SOURce<hw>]:BB:BT00th:PCONfiguration:DLENght.....	124
[:SOURce<hw>]:BB:BT00th:PCONfiguration:DSFPacket.....	124
[:SOURce<hw>]:BB:BT00th:PCONfiguration:DWHitening.....	125
[:SOURce<hw>]:BB:BT00th:PCONfiguration:EIROPacketfollows.....	125
[:SOURce<hw>]:BB:BT00th:PCONfiguration:HFCControl.....	126

[:SOURce<hw>]:BB:BT00th:PCONfiguration:LFSWord.....	126
[:SOURce<hw>]:BB:BT00th:PCONfiguration:LTAddress.....	126
[:SOURce<hw>]:BB:BT00th:PCONfiguration:PFControl.....	127
[:SOURce<hw>]:BB:BT00th:PCONfiguration:PLENght.....	127
[:SOURce<hw>]:BB:BT00th:PCONfiguration:SLAP.....	128
[:SOURce<hw>]:BB:BT00th:PCONfiguration:SNSValue.....	128
[:SOURce<hw>]:BB:BT00th:PCONfiguration:SRMode.....	128
[:SOURce<hw>]:BB:BT00th:PCONfiguration:VDAta.....	129
[:SOURce<hw>]:BB:BT00th:PFORmat.....	131
[:SOURce<hw>]:BB:BT00th:PRAMPing:FOFFset.....	192
[:SOURce<hw>]:BB:BT00th:PRAMPing:RFUNction.....	192
[:SOURce<hw>]:BB:BT00th:PRAMPing:ROFFset.....	193
[:SOURce<hw>]:BB:BT00th:PRAMPing:RTIME.....	193
[:SOURce<hw>]:BB:BT00th:PRESet.....	109
[:SOURce<hw>]:BB:BT00th:PTYPE.....	119
[:SOURce<hw>]:BB:BT00th:SETTing:CATalog.....	109
[:SOURce<hw>]:BB:BT00th:SETTing:DELeTe.....	110
[:SOURce<hw>]:BB:BT00th:SETTing:LOAD.....	110
[:SOURce<hw>]:BB:BT00th:SETTing:STORe.....	110
[:SOURce<hw>]:BB:BT00th:SLENght.....	119
[:SOURce<hw>]:BB:BT00th:SRATe:VARiation.....	191
[:SOURce<hw>]:BB:BT00th:STATe.....	110
[:SOURce<hw>]:BB:BT00th:STIMing.....	120
[:SOURce<hw>]:BB:BT00th:TMODe.....	111
[:SOURce<hw>]:BB:BT00th:TRIGger:ARM:EXECute.....	198
[:SOURce<hw>]:BB:BT00th:TRIGger:EXECute.....	198
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:DELay.....	200
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:FESHift.....	200
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:MODE.....	201
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:OFFTime.....	201
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:ONTime.....	201
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:PATtern.....	202
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:PINdex.....	202
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:PULSe:DIVider.....	202
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:PULSe:FREQUency?.....	202
[:SOURce<hw>]:BB:BT00th:TRIGger:OUTPut<ch>:RESHift.....	203
[:SOURce<hw>]:BB:BT00th:TRIGger:RMODe.....	196
[:SOURce<hw>]:BB:BT00th:TRIGger:SLENght.....	197
[:SOURce<hw>]:BB:BT00th:TRIGger:SLUNit.....	197
[:SOURce<hw>]:BB:BT00th:TRIGger:SOURce.....	195
[:SOURce<hw>]:BB:BT00th:TRIGger:TIME:DATE.....	196
[:SOURce<hw>]:BB:BT00th:TRIGger:TIME:TIME.....	196
[:SOURce<hw>]:BB:BT00th:TRIGger:TIME[:STATe].....	197
[:SOURce<hw>]:BB:BT00th:TRIGger[:EXTernal]:DELay.....	198
[:SOURce<hw>]:BB:BT00th:TRIGger[:EXTernal]:INHibit.....	199
[:SOURce<hw>]:BB:BT00th:TRIGger[:EXTernal]:SYNChronize:OUTPut.....	198
[:SOURce<hw>]:BB:BT00th:UNIT:TIME.....	111
[:SOURce<hw>]:BB:BT00th:UPTYpe.....	132
[:SOURce<hw>]:BB:BT00th:USLEnght.....	133
[:SOURce<hw>]:BB:BT00th:VERSion?.....	111

[:SOURce<hw>]:BB:BT0oth:WAVeform:CREate.....	112
[:SOURce<hw>]:BB:BT0oth[:TRIGger]:SEQUence.....	195

# Index

## Symbols

$\pi/4$  DQPSK/8DPSK filter ..... 92

## A

Access address ..... 63  
 ACK ..... 44  
 Acknowledgment ..... 44  
 ACL+EDR ..... 34  
 Advertisers address type ..... 66  
 Advertisers device address ..... 68  
 Advertising channel table ..... 61  
 Advertising event  
   Settings ..... 56  
 Advertising event delay ..... 56  
 Advertising event interval ..... 56  
 Advertising packet interval ..... 57  
 Advertising settings  
   Advertising event delay ..... 56  
   Advertising packet interval ..... 57  
   Scan interval ..... 57  
   Scan window ..... 57  
   Test packet interval ..... 88  
   Test packet payload length ..... 89  
   Transmit window offset ..... 57, 74  
   Transmit window size ..... 58, 73  
 All data ..... 43  
 Antenna gain ..... 66  
 Antenna number ..... 66  
 Application cards ..... 10  
 Application notes ..... 10  
 Arm ..... 100  
 Armed  
   Auto, trigger mode ..... 98  
   Retrigger, trigger mode ..... 98  
 Auto  
   Trigger mode ..... 98

## B

Baseband clipping ..... 95  
 Baseband filter ..... 91, 92  
 Basic rate+EDR  
   Bluetooth ..... 34  
 Bluetooth  
   Basic rate+EDR ..... 34  
   Dirty transmitter test ..... 35  
   Filter/clipping ..... 35  
   Low energy ..... 34  
   Mode ..... 34  
   Power ramping ..... 35  
   Settings ..... 32  
   Version ..... 34  
 Bluetooth basic rate+EDR  
   Settings ..... 40  
 Bluetooth LE  
   Angle of arrival ..... 27  
   Angle of departure ..... 28  
   Direction finding ..... 27  
 Bluetooth low energy  
   Channel settings ..... 49  
 Brochures ..... 10

BxT ..... 91, 92

## C

Carrier frequency offset (DTT) ..... 39  
 Center frequency ..... 61  
   Secondary advertising channel table ..... 84  
 Channel index ..... 61  
   Secondary advertising channel table ..... 84  
 Channel settings  
   Channel type ..... 51  
   Controller role ..... 53  
   Controller state ..... 53  
   CRC ..... 53  
   Duration ..... 54  
   Duty cycle ..... 52  
   Event configuration ..... 54  
   Frame configuration ..... 54  
   Modulation format ..... 54  
   Packet format ..... 52  
   Packet type ..... 51  
   Payload type ..... 54  
   Sequence length ..... 53  
   Test packet configuration ..... 54  
 Channel state ..... 61  
   Secondary advertising channel table ..... 84  
 Channel table  
   Settings ..... 61  
 Channel type ..... 61  
   Secondary advertising channel table ..... 84  
 Class of Device ..... 47  
 Clipping  
   Level ..... 95  
   Mode ..... 95  
   Settings ..... 94  
   State ..... 95  
 Clipping Mode ..... 186  
 Clock  
   Source ..... 105  
 Clock accuracy  
   Secondary advertising channel table ..... 84  
 Company ID ..... 75  
 Connection event interval ..... 59, 71  
 Connection settings  
   Connection event interval ..... 59, 71  
   LL connection mode ..... 59  
   Long-term key (hex) ..... 60  
   No. of Tx packets per event ..... 58  
   Selected data channel ..... 60  
 Controller role  
   Advertiser ..... 53  
   Central ..... 53  
   Peripheral ..... 53  
   Scanner ..... 53  
 Conventions  
   SCPI commands ..... 106  
 CRC initial ..... 65  
 Crest factor ..... 95  
 CTE info ..... 65, 76  
 CTE time ..... 65  
 CTE type ..... 66  
 Current range without recalculation ..... 104  
 Cut off frequency factor ..... 93

**D**

Data channel table .....	61
Data event connection	
Settings .....	58
Data length	
Packet configuration .....	46, 48, 71
Data sheets .....	10
Data source	
Packet configuration .....	45, 48, 70
Voice Field .....	47
Data whitening .....	43, 63
Data Whitening .....	150
Default settings .....	33
Delay	
Marker .....	104
Trigger .....	101
Device address .....	43, 68
Devices address type .....	66
Direction finding .....	27
Dirty transmitter test .....	37
Bluetooth .....	35
Carrier frequency offset .....	39
Frequency drift deviation .....	38
Frequency drift rate .....	38
Modulation index .....	39
Number of packets per set .....	38
Set to default .....	37
Settings .....	35
Start phase .....	37
State (long/short table) .....	38
Symbol timing error .....	39
Documentation overview .....	9

**E**

EIR packet follows .....	46
Encrypted DIVERSIFIER(hex) .....	73
Encrypted mode .....	59
Error code .....	74
eSCO+EDR .....	34
Event active part	
Marker .....	102
Event configuration	
Channel table .....	61
Event inactive part	
Marker .....	102
Event start	
Marker .....	102
Event/frame configuration	
Settings Bluetooth LE .....	54

**F**

Fall offset .....	97
Feature set length .....	73
Filter .....	91, 92
Settings .....	91
Filter parameter	
Roll off .....	91, 92
Filter/clipping	
Bluetooth .....	35
Filtering, clipping	
Settings .....	90
Flow control (header) .....	44
Flow control (Payload) .....	46, 48

Frame active part	
Marker .....	102
Frame configuration	
Channel table .....	61
Settings .....	56
Frame inactive part	
Marker .....	102
Frame start	
Marker .....	102
Frequency deviation .....	93
Frequency drift deviation .....	38
Frequency drift rate .....	38

**G**

Gauss FSK filter .....	91
General settings	
Accessing .....	32
Save/Recall .....	34
Set to default .....	33
State .....	33
Generate	
Waveform file (Bluetooth) .....	34
Getting started .....	9

**H**

Help .....	9
Hide data channel table .....	72
Hide data channel-mapping table .....	72
Hop length .....	72

**I**

Impulse length .....	92
Inhibit	
External trigger .....	101
Initialization vector (hex) .....	73
Initiators address type .....	66
Initiators device address .....	68
Installation .....	7
Instrument help .....	9
Instrument security procedures .....	9

**K**

Key features .....	7
--------------------	---

**L**

LAP .....	43
LAP for sync word .....	43
Level	
Clipping .....	95
LL connection event count .....	72
LL connection mode .....	59
LL connection timeout .....	71
Logical transport address .....	44
Long-term key (hex) .....	60
Low energy	
Bluetooth .....	34

**M**

Marker	
Delay .....	104
Mode .....	102
MIC (message authentication code) .....	59

Mode		
Bluetooth	34	
Clipping	95	
Modulation		
Settings	93	
Modulation index	94	
Modulation index (DTT)	39	
Modulation type	93	
<b>N</b>		
NAK	44	
NAP	43	
NESN start value	64	
No. of Tx packets per event	58	
Number of packets per set	38	
<b>O</b>		
Open source acknowledgment (OSA)	10	
Oversampling	92	
<b>P</b>		
Packet configuration	58	
Access address	63	
Acknowledgment	44	
Advertisers address type	66	
Antenna gain	66	
Antenna number	66	
Bluetooth basic rate+EDR	41	
Bluetooth LE	62	
Center frequency	61	
Channel index	61	
Channel state	61	
Channel type	61	
Class of device	47	
Company ID	75	
Connection event interval	59, 71	
CRC initial	65	
CTE info	65, 76	
CTE time	65	
CTE type	66	
Data whitening	43, 63	
Device address	43	
EIR packet follows	46	
Encrypted DIVERSifier(hex)	73	
Error code	74	
Feature set length	73	
Flow control (header)	44	
Flow control (Payload)	46, 48	
Hide data channel table	72	
Hide data channel-mapping table	72	
Hop length	72	
Initialization vector (hex)	73	
Initiators address type	66	
Initiators device address	68	
LAP	43	
LAP for sync word	43	
LL connection event count	72	
LL connection timeout	71	
Logical transport address	44	
NAP	43	
NESN start value	64	
Packet Length	49	
Peripheral latency	71	
Random vector (hex)	72	
Scan repetition mode	46	
Scanners device address	68	
SEQN start value	45	
Session key if (hex)	73	
Show data channel mapping table	72	
Show data channel table	72	
Simulation mode	43	
Sleep clock accuracy	74	
SN start value	64	
Subversion number	75	
Synchronize LAP with sync word	43	
UAP	43	
Unknown type (hex)	71	
Version number	75	
Packet Configuration	41	
Packet Configuration:Data Whitening	150	
Packet editor	43	
Packet Length	49	
Packet structure	83	
Packet Type	40	
Payload type test packet	89	
Periodic advertising interval	56	
Peripheral latency	71	
Power ramping		
Bluetooth	35	
Settings	96	
Private address type	66, 68	
Public address type	66, 68	
Pulse		
Marker	102	
<b>R</b>		
Raised cosine filter		
see Cosine filter	91	
Ramp function	96	
Ramp Function	192	
Ramp time	96	
Ramp Time	193	
Random vector (hex)	72	
Release notes	10	
Remote control		
Programming examples	106	
Restart		
Marker	102	
Retrigger		
Trigger mode	98	
Rise offset	96	
Roll off factor	91, 92	
Root raised cosine filter		
see Root Cosine	91	
RRC filter		
see Root Cosine filter	91	
<b>S</b>		
Safety instructions	10	
Scan interval	57	
Scan repetition mode	46	
Scan window	57	
Scanners device address	68	
SCO	34	
Secondary advertising channel table	84	
Security procedures	9	
Select data channel index	60	
Selected data channel	60	
SEQN start value	45	

Sequence Length .....	40
Service manual .....	9
Session key id (hex) .....	73
Set to default	
Dirty transmitter test .....	37
Settings	
Advertising event .....	56
Advertising event interval .....	56
Bluetooth .....	32
Bluetooth basic rate+EDR .....	40
Channel table .....	61
Clipping .....	94
Data event connection .....	58
Dirty transmitter test .....	35
Event /frame configuration Bluetooth LE .....	54
Filter .....	91
Filter, clipping .....	90
Frame configuration .....	56
Modulation .....	93
Packet configuration, basic rate+EDR .....	41
Packet configuration, Bluetooth LE .....	62
Periodic advertising interval .....	56
Power ramping .....	96
Test packet configuration .....	88
Trigger .....	97
Show data channel mapping table .....	72
Show data channel table .....	72
Signal duration unit .....	99
Signal generation status .....	99
Simulation mode .....	43
Single	
Trigger .....	98
Sleep clock accuracy .....	74
Slot timing .....	41
SN start value .....	64
Source	
Clock .....	105
Trigger .....	100
Standard settings .....	33
Start phase .....	37
State	
(long table) .....	38
(short table) .....	38
Clipping .....	95
Subversion number .....	75
Symbol rate variation .....	94
Symbol timing error (DTT) .....	39
Synchronize LAP with sync word .....	43
<b>T</b>	
Test packet configuration	
Settings .....	88
Test packet interval .....	88
Test packet payload length .....	89
Test packet payload type .....	89
Time-based trigger	
Date .....	99
State .....	99
Time .....	99
Transmit window offset .....	57, 74
Transmit window size .....	58, 73
Transport Mode .....	34
Transport Mode - ACL+EDR .....	34
Transport Mode - eSCO+EDR .....	34
Transport Mode - SCO .....	34
Trigger	
Delay .....	101
Execute .....	100
External .....	101
External, inhibit .....	101
Mode .....	98
Signal duration .....	99
Source .....	100
Synchronize output .....	100
Trigger settings .....	97
<b>U</b>	
UAP .....	43
Unencrypted .....	59
Unknown type (hex) .....	71
User manual .....	9
<b>V</b>	
Version	
Bluetooth .....	34
Version number .....	75
Videos .....	10
<b>W</b>	
Waveform file	
Create (Bluetooth) .....	34
White papers .....	10