

# R&S®SMW-K89

## NFC A/B/F

## User Manual



1176787802  
Version 15

**ROHDE & SCHWARZ**  
Make ideas real



This document describes the following software option:

- R&S®SMW-K89 NFC A/B/F (1413.6619.xx)

This manual describes firmware version FW 5.30.047.xx and later of the R&S®SMW200A.

© 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.

EMV® is a registered trademark in the U.S. and other countries and an unregistered trademark elsewhere. The EMV trademark is owned by EMVCo, LLC.

NFC Forum™ and the NFC Forum logo are trademarks of the Near Field Communication Forum.

All other trademarks are the properties of their respective owners.

1176.7878.02 | Version 15 | R&S®SMW-K89

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

# Contents

<b>1</b>	<b>Welcome to the NFC/EMV digital standard.....</b>	<b>7</b>
1.1	Accessing the NFC dialog.....	7
1.2	What's new.....	8
1.3	Documentation overview.....	8
1.3.1	Getting started manual.....	8
1.3.2	User manuals and help.....	8
1.3.3	Tutorials.....	9
1.3.4	Service manual.....	9
1.3.5	Instrument security procedures.....	9
1.3.6	Printed safety instructions.....	9
1.3.7	Data sheets and brochures.....	9
1.3.8	Release notes and open source acknowledgment (OSA).....	9
1.3.9	Application notes, application cards, white papers, etc.....	10
1.3.10	Videos.....	10
1.4	Scope.....	10
1.5	Notes on screenshots.....	10
<b>2</b>	<b>Required options.....</b>	<b>12</b>
<b>3</b>	<b>About the NFC digital standard.....</b>	<b>13</b>
3.1	Basics of data transmission with NFC.....	13
3.1.1	Power transmission and data transmission from a polling device.....	14
3.1.2	Data transmission from a listening device.....	14
3.1.3	Modulation scheme and coding.....	15
3.1.4	NFC operating modes, modulation and coding.....	17
3.2	Timing aspects.....	18
3.3	Leveling aspects.....	20
3.3.1	Interpretation of "RF level" indication.....	20
3.3.2	Desired voltage of the unmodulated signal.....	20
<b>4</b>	<b>About the EMV contactless digital standard.....</b>	<b>22</b>
4.1	Basics of data transmission with EMV contactless.....	22
4.2	EMV contactless transmission technologies, modulation and coding.....	23

<b>5</b>	<b>NFC configuration and settings.....</b>	<b>24</b>
5.1	General settings.....	24
5.2	Trigger settings.....	28
5.3	Marker settings.....	33
5.4	Clock settings.....	35
5.5	Local and global connectors settings.....	36
5.6	Modulation settings.....	37
5.6.1	Common settings.....	37
5.6.2	Slope settings.....	41
5.7	Predefined sequence.....	43
5.8	Sequence configuration settings.....	44
5.9	Frame configuration settings.....	50
5.9.1	Parameters of the frame configuration dialog.....	50
5.10	Clipping settings.....	67
<b>6</b>	<b>How to generate signals with the NFC A/B/F option.....</b>	<b>68</b>
6.1	How to generate a signal for test of an NFC device in polling mode.....	68
6.2	How to generate a signal for test of an NFC device in listener mode.....	69
<b>7</b>	<b>Remote-control commands.....</b>	<b>72</b>
7.1	Programming example.....	73
7.2	Primary settings.....	76
7.3	Save/recall operations.....	78
7.4	Predefined sequence.....	79
7.5	Sequence configuration.....	81
7.6	Frame configuration.....	84
7.7	Modulation settings.....	112
7.8	Clipping settings.....	115
7.9	Trigger settings.....	116
7.10	Marker settings.....	121
7.11	Clock settings.....	123
	<b>Glossary: List of terms and abbreviations.....</b>	<b>125</b>
	<b>List of commands.....</b>	<b>131</b>
	<b>Index.....</b>	<b>136</b>





# 1 Welcome to the NFC/EMV digital standard

The R&S SMW-K89 is a firmware application that adds functionality to generate signals in accordance with the NFC/EMV.

The R&S SMW-K89 NFC features:

- Supports NFC- A, NFC- B and NFC- F.
- Supports poll and listen transmission mode.
- Offers a quick frame configuration through a selection of predefined sequences for the poll transmission mode.
- Offers a flexible configuration of the frame sequence.
- Supports all relevant command types.

The R&S SMW-K89 EMV features:

- Supports EMV Type A and EMV Type B.
- Supports PICC to PCD and PCD to PICC transmission mode.
- Offers a quick frame configuration through a selection of predefined sequences for the PCD to PICC transmission mode.
- Offers a flexible configuration of the frame sequence.
- Supports all relevant command types.

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 SMW user manual. The latest version is available at:

[www.rohde-schwarz.com/manual/SMW200A](http://www.rohde-schwarz.com/manual/SMW200A)

## Installation

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

## 1.1 Accessing the NFC dialog

### To open the dialog with NFC settings

- In the block diagram of the R&S SMW, select "Baseband > NFC/EMV".

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".

## 1.2 What's new

This manual describes firmware version FW 5.30.047.xx and later of the R&S®SMW200A.

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

- Time-based triggering, see "[Time Based Trigger](#)" on page 30 and "[Trigger Time](#)" on page 30.
- Editorial changes

## 1.3 Documentation overview

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

[www.rohde-schwarz.com/manual/smw200a](http://www.rohde-schwarz.com/manual/smw200a)

### 1.3.1 Getting started manual

Introduces the R&S SMW 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.

### 1.3.2 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 SMW is not included.

The contents of the user manuals are available as help in the R&S SMW. 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.



### 1.3.3 Tutorials

The R&S SMW provides interactive examples and demonstrations on operating the instrument in form of tutorials. A set of tutorials is available directly on the instrument.

### 1.3.4 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>

### 1.3.5 Instrument security procedures

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

### 1.3.6 Printed safety instructions

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

### 1.3.7 Data sheets and brochures

The data sheet contains the technical specifications of the R&S SMW. 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/smw200a](http://www.rohde-schwarz.com/brochure-datasheet/smw200a)

### 1.3.8 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/smw200a](http://www.rohde-schwarz.com/firmware/smw200a)

### 1.3.9 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/smw200a](http://www.rohde-schwarz.com/application/smw200a) and [www.rohde-schwarz.com/manual/smw200a](http://www.rohde-schwarz.com/manual/smw200a)

### 1.3.10 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.



Figure 1-1: Product search on YouTube

## 1.4 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 SMW user manual.

## 1.5 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 Required options

The basic equipment layout for generating NFC signals includes:

- Standard or wideband baseband generator (R&S SMW-B10/-B9)
- Baseband main module (R&S SMW-B13/-B13T) or wideband baseband main module (R&S SMW-B13XT)
- Frequency option (e.g. R&S SMW-B1003)  
(per signal path)
- Digital standard NFC A/B/F (R&S SMW-K89)  
(per signal path)

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

For more information, see data sheet.

## 3 About the NFC digital standard

The following description is taken from the R&S White Paper 1MA182: "Near Field Communication (NFC) Technology and Measurements" which contains further practical hints.

Near Field Communication (NFC) is a new, short-range wireless connectivity technology that evolved from a combination of existing contactless identification and interconnection technologies. It was jointly developed by Sony and NXP Semiconductors (formerly Philips).

NFC is designed to enable the exchange of various types of information, such as telephone numbers, pictures, MP3 files or digital authorizations between two NFC enabled devices like mobile phones, or between an NFC enabled mobile phone and a compatible RFID chip card or reader that are held close to each other. NFC is intended to be used as an access key to contents and for services such as cashless payment, ticketing and access control.

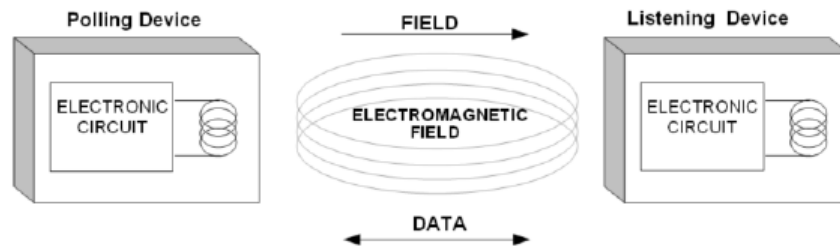
NFC operates in a frequency range centered on 13.56 MHz and offers a data transmission rate of up to 424 kbit/s within a distance of approximately 10 centimeters. In contrast to the conventional contactless technology in this frequency range (only active-passive communications), communications between NFC-capable devices can be active-active (peer-to-peer) as well as active-passive, NFC therefore represents a link to the RFID world. NFC is backwards compatible with the widely used Smart Card infrastructure based on ISO/IEC 14443 A (e. g. NXP's MIFARE technology) and ISO/IEC 14443 B as well as with the Sony FeliCa card (JIS X 6319-4). For the exchange of information between two NFC devices, a new protocol was developed which is defined in the standards ECMA-340 and ISO/IEC 18092.

To guarantee the function of NFC devices conforming to the standards as well as comprehensive protocol tests, a number of RF tests also have to be carried out. An NFC generator is an essential part of these tests. The option R&S SMW-K89 enables you to generate signals in accordance with the NFC standard.

The NFC specific abbreviations used in this manual as well as the different types of tag platforms/protocols (e.g. Type 4A Tag, NFC-DEP) are described in the NFC Digital Protocol Technical Specification. All mentioned standards are available under [www.nfc-forum.org](http://www.nfc-forum.org).

### 3.1 Basics of data transmission with NFC

Like the RFID Standards 14443 and FeliCa NFC uses an inductive coupling. Similar to the transformer principle, the magnetic near-field of two conductor coils is used to couple the polling device (initiator) and listening device (target).



*Figure 3-1: Polling device (initiator) and listening device (target) configuration*

The operating frequency is 13.56 MHz, and a bitrate of 106 kbit/s (partly also 212 kbit/s and 424 kbit/s) is used. Modulation schemes are amplitude on/off keying (OOK) with different modulation depth (100 % or 10 %) and BPSK.

### 3.1.1 Power transmission and data transmission from a polling device

For transmission to a passive system such as an NFC phone in passive card emulation mode, the passive system uses the 13.56 MHz carrier signal of the polling device as energy source. Modulation scheme of the polling device is ASK. For NFC peer-to-peer mode, both directions are modulated and coded like a polling device. However less power is necessary because both NFC devices use their own power supply and the carrier signal is switched off after end of transmission.

### 3.1.2 Data transmission from a listening device

Due to the coupling of the coils of a polling and a listening device, a passive listening device also affects the active polling device. A variation in the impedance of the listening device causes amplitude changes to the antenna voltage of the polling device, detected by the polling device. This technique is called load modulation. Load modulation is carried out in listening mode (as with ISO/IEC 14443) using an auxiliary carrier at 848 kHz which is modulated by the baseband and varies the impedance of the listening device. The [Figure 3-2](#) shows the spectrum with load modulation. Modulation spectra of carrier and auxiliary carriers are indicated with triangles (Modulation spectra of carrier and of auxiliary carriers do not appear at the same time because NFC uses time division multiplexing). The modulation scheme is ASK (as with ISO/IEC 14443 A PICC's) or BPSK as with 14443 B PICC's. There is a third passive mode which is compatible to FeliCa where the load modulation is without an auxiliary carrier directly as ASK on the 13.56 MHz carrier.

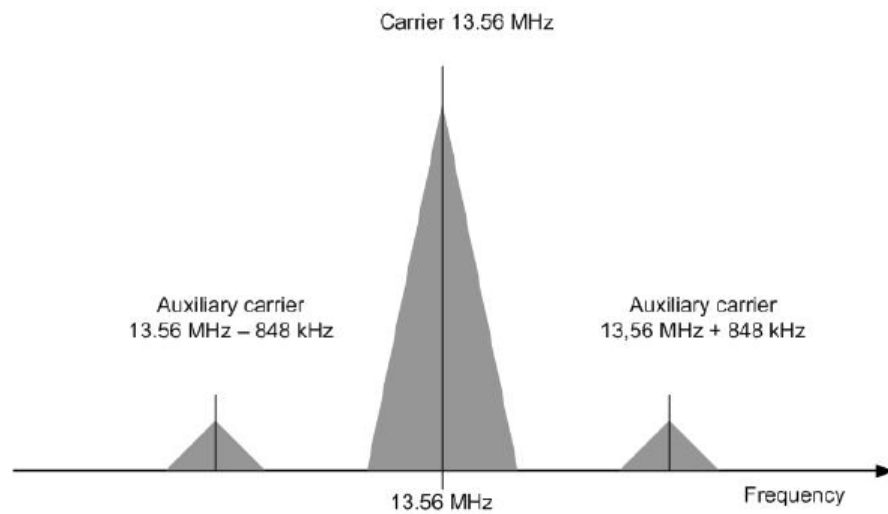


Figure 3-2: Load modulation on a 13.56 MHz carrier with 848 kHz auxiliary carrier.

### 3.1.3 Modulation scheme and coding

Amplitude shift keying (OOK) with different modulation depths (100% or 10%) or BPSK (as with ISO/IEC 14443 B PICC's) is used.

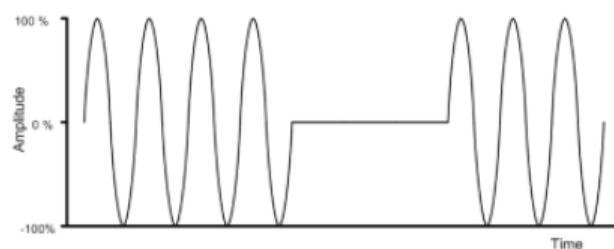


Figure 3-3: ASK with 100% modulation depth

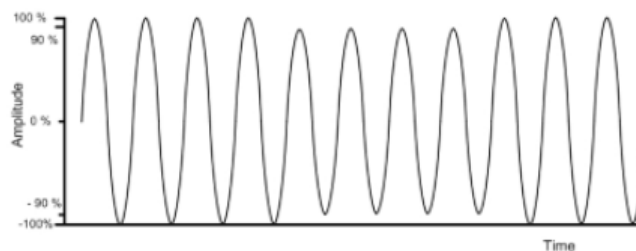


Figure 3-4: ASK with 10% modulation depth

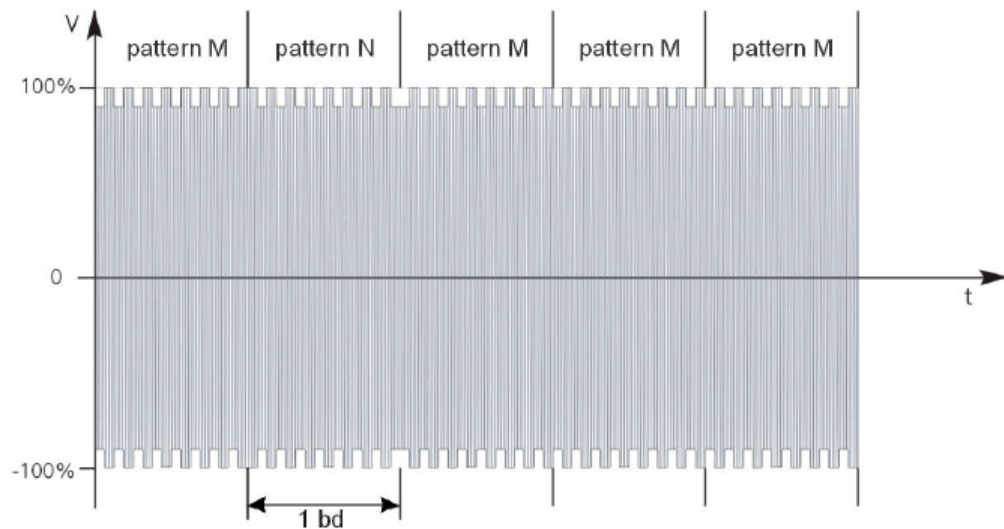


Figure 3-5: NRZ-L Coding with BPSK

NRZ-L, Modified Miller and Manchester Coding are used by NFC.

- With NRZ-L a “high”-state during a bit duration indicates a logic 1, a “low”-state a logic 0.
- With Manchester Coding the first half of a bit will be set to “high”-state at a logic 1, and the second half to “low state”. With a logic 0, the first half of a bit is set to “low”-state and the second half to “high”-state.
- With Modified Miller Coding with a logic 1 a “low” pulse occurs after half of the bit duration. With a logic 0 a “low”-pulse occurs at the beginning of a bit. Exception: If a logic 0 follows a 1 no pulse occurs, the signal remains high.

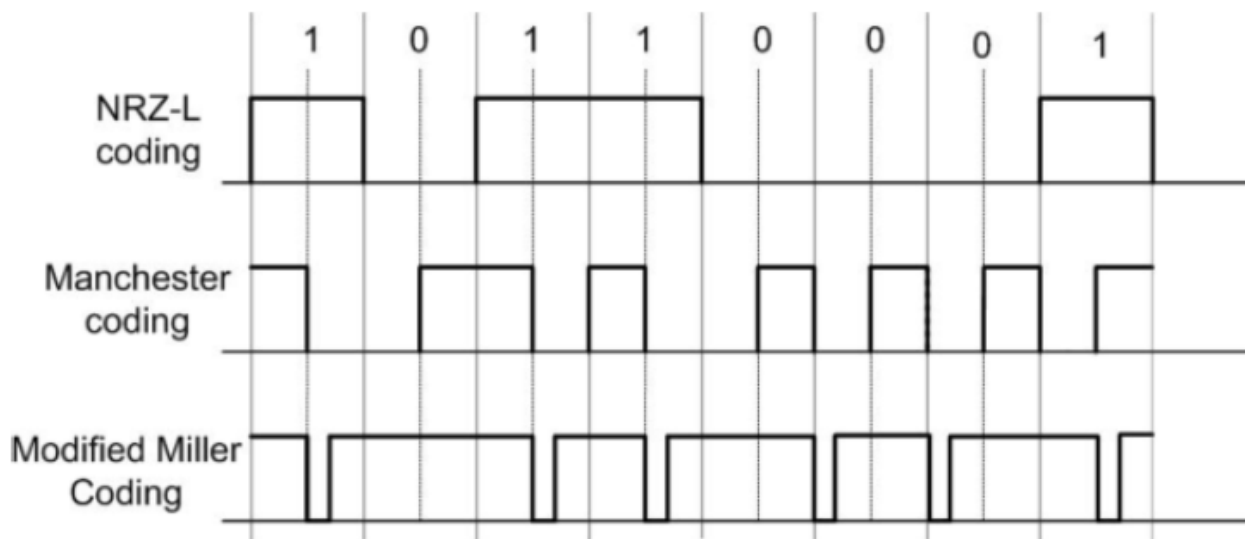


Figure 3-6: Coding with NFC is either NRZ\_L, Modified Miller or Manchester



In [Figure 3-7](#) load modulation is visualized for ASK modulation with Manchester Coding (14443 A PICC or NFC-A device in passive card emulation mode, see [Chapter 3.1.4, "NFC operating modes, modulation and coding"](#), on page 17)

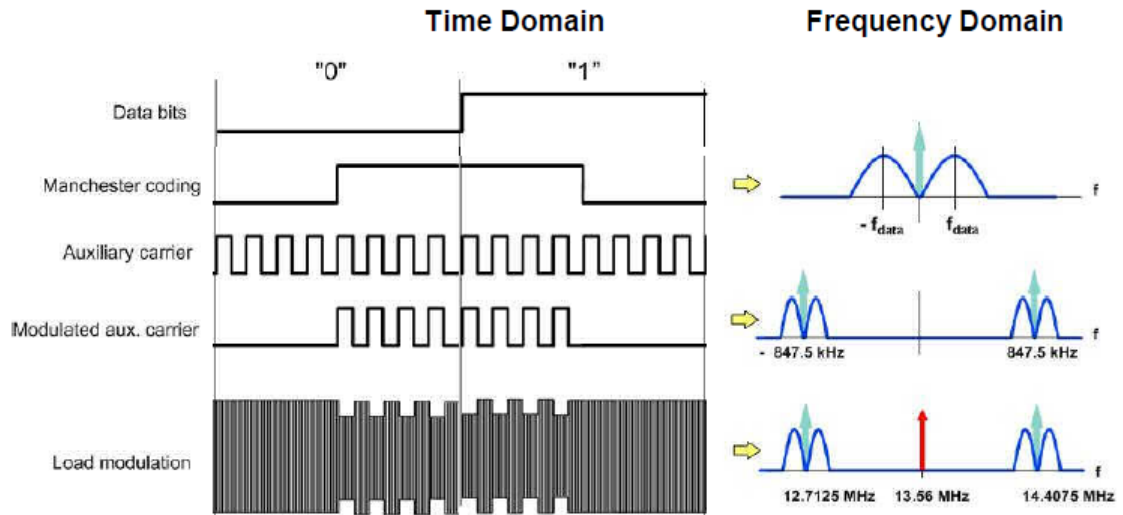


Figure 3-7: Visualization of load modulation with auxiliary carrier in time and frequency domain

### 3.1.4 NFC operating modes, modulation and coding

There are three main operating modes for NFC:

- Card emulation mode (passive mode): the NFC device behaves like an existing contactless card conforming to one of the legacy standards
- Peer-to-peer mode: two NFC devices exchange information. The initiator device (polling device) requires less power compared to the reader/writer mode because the target (listener) uses its own power supply.
- Reader/writer mode (active mode): the NFC device is active and reads or writes to a passive legacy RFID tag.

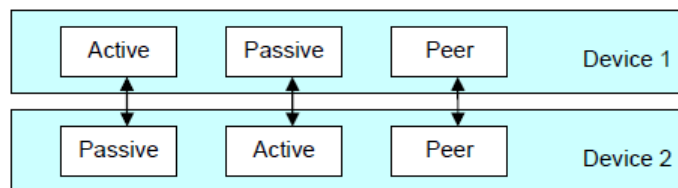


Figure 3-8: NFC operating modes

Every mode (card emulation, peer-to-peer, reader/writer mode) can be combined with one of the following transmission technologies:

- NFC-A (backward compatible to ISO/IEC 14443 A)
- NFC-B (backward compatible to ISO/IEC 14443 B)
- NFC-F (backward compatible to JIS X 6319-4)

To support all the different technologies, an NFC device in polling mode first attempts to get responses from NFC-A, NFC-B and NFC-F tags with the according request signals. When getting a response from a compatible device, the NFC device sets up the corresponding communication mode (NFC-A, NFC-B or NFC-F mode).

Coding and modulation varies depending on active or passive communication mode, NFC-A, -B, -F communication, and bitrate.

The [Table 3-1](#) shows coding, modulation and data rates for NFC-A, -B or -F communication.

**Table 3-1: NFC RF standards overview**

NFC forum Standard	Polling / Listening	Coding	Modulation	Data rate	Carrier frequency
NFC-A	Polling	Modified Miller	ASK 100%	106 kb/s	13.56 MHz
	Listening	Manchester	Load modulation (ASK subcarrier)	106 kb/s	13.56 MHz +-848kHz subcarrier
NFC-B	Polling	NRZ-L	ASK 10%	106 kb/s	13.56 MHz
	Listening	NRZ-L	Load modulation (BPSK subcarrier)	106 kb/s	13.56 MHz+-848kHz subcarrier
NFC-F	Polling	Manchester	ASK 10%	212 / 424 kb/s	13.56 MHz
	Listening	Manchester	Load modulation (APSK)	212 / 424 kb/s	13.56 MHz (without subcarrier)

## 3.2 Timing aspects

The NFC specification defines the duration of the individual commands as a number of bits. This instrument generates the signal as sample sequence where the applied sample rate is user-defined. Depending on the selected sampling rate it may be that the duration of a command expressed in samples does not result in an integer number of samples. In this implementation however the length of the sequence is always an integer number of samples, i.e. the software rounds up the number of samples to the next integer value. The rounding up procedure is applied on command basis, even if a command is repeated.

The [Figure 3-10](#) shows this principle as an example.

### Example:

The [Figure 3-9](#) shows an example of a sequence with the following settings:

- Sample Rate = 20.1 Msps
- [Technology](#) > "NFC-A"
- [Transmission Mode](#) > "Poll"

Command Type	Rep.	Duration (µs)
"SENS_REQ"	1	(calculated and displayed automatically)
"IDLE"	1	0.05

Command Type	Rep.	Duration (µs)
"ALL_REQ"	2	(calculated and displayed automatically)
"BLANK"	1	0.1

NFC/EMV

General **Stop** Trigger In Auto Marker Clock Internal Modulation Predefined Sequence **Sequence**

0x26  
7 bits

Total Sequence Duration: 255.174 µs    Total Number of Samples: 5 129

	Start Time (us)	Command Type	Rep.	Power Offset (dB)	Duration (us)	Samples	Frame Conf.
1 >	0.000	SENS_REQ	1	0.00	84.96	1 708	Config...
2	84.975	IDLE	1	0.00	0.05	2	Config...
3	85.075	ALL_REQ	2	0.00	169.91	3 416	Config...
4	255.025	BLANK	1	0.00	0.10	3	Config...

Append Insert Delete Copy Paste

Figure 3-9: Example of sequence configuration settings

The Figure 3-10 illustrates the calculation of the Start Time per command and the parameters Total Sequence Duration and Total Number of Samples.

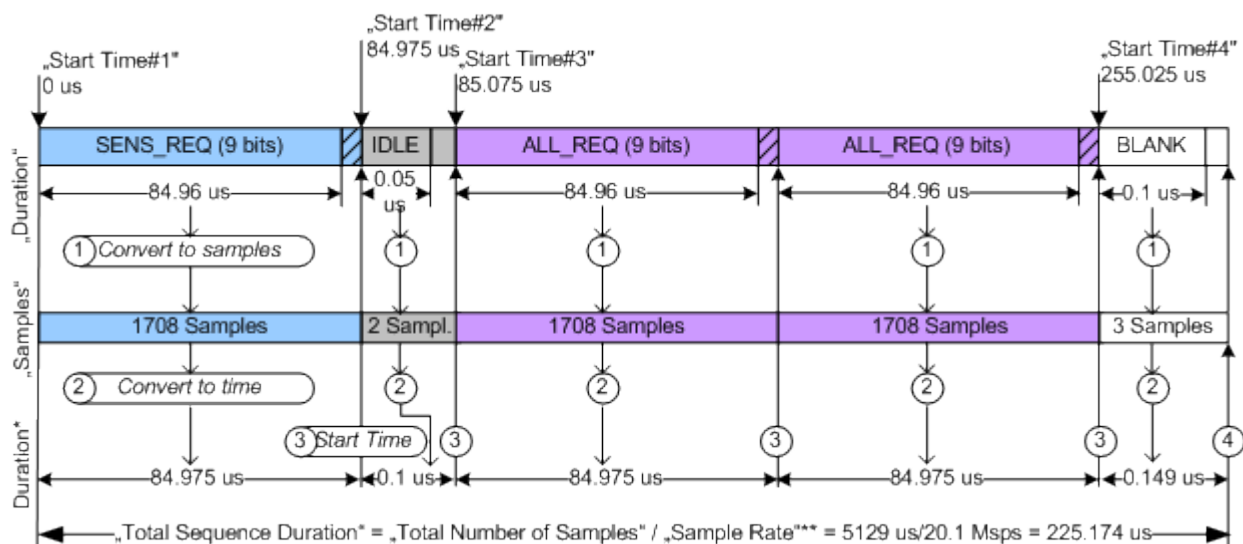


Figure 3-10: Calculation of duration and number of samples for Sample Rate = 20.1 Msps

- 1 = "Duration" \* "Sample Rate" = # Samples, "Samples" = Cell (# samples)
- 2 = Duration\* = "Samples"/"Sample Rate"
- 3 = "Start Time"<sub>N</sub> = (Duration\*<sub>1</sub> + .. + Duration\*<sub>N-1</sub>) = ("Samples"<sub>1</sub> + ... + "Samples"<sub>N-1</sub>)/"Sample Rate"
- 4 = "Total Number of Samples" = "Samples"<sub>1</sub> + ... + "Samples"<sub>N</sub>



For some modulation settings, especially for those that cause very smooth signal edges, it might be necessary that the implementation not only rounds up to the next integer number of samples, but also that it enlarges the commands even more, in order to prevent a sharp cutting of the last signal edge of the command.

### 3.3 Leveling aspects

This chapter describes general leveling aspects.

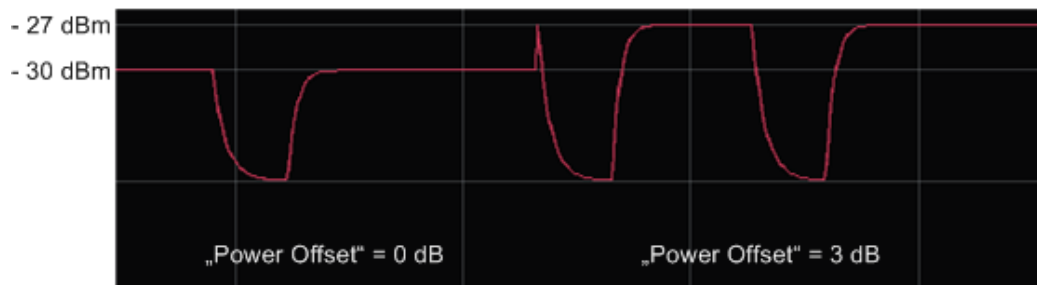
#### 3.3.1 Interpretation of "RF level" indication

This chapter describes the interpretation of the RF Level.



**The "RF Level" indication of the generators does not display the RMS signal level!**

The "RF Level" indication in the header of the instrument refers to the power during the unmodulated parts of the signal, i.e. the part where the relative signal voltage is 100% (outside of overshoots) and the "Power Offset" is 0 dB (see [Figure 3-11](#)).



*Figure 3-11: Signal leveling of a sequence build from two commands with "Power Offset" of 0 dB and 3 dB respectively, "RF Level" = -30dBm and "PEP" = -27 dBm*

#### 3.3.2 Desired voltage of the unmodulated signal

Several test cases require a listener test signal prior to the modulation on the RF carrier. This test signal is then supplied to the reference listener antenna. Three leveling parameters are provided to calculate the settings to reach the required voltage of the unmodulated signal automatically.

These parameters are available for "Transmission Mode > Listen" and "NFC State > On", for details see ["Unmodulated Parts Voltage To Peak Voltage Ratio"](#) on page 27 ,

["Desired Voltage In Unmodulated Signal parts"](#) on page 27 and ["Update Analog I/Q Settings For Desired Voltage"](#) on page 27.

To use these leveling parameters...

1. Define the "Desired Voltage In Unmodulated Signal Parts".
2. Select "Update Analog I/Q Settings For Desired Voltage" to automatically adjust the settings at the I/Q output connectors ("I/Q Level Vp (EMF)").

Parameter "Unmodulated Parts Voltage To Peak Voltage Ratio" displays the ratio of the voltage in the unmodulated parts of the signal to its peak value.

## 4 About the EMV contactless digital standard

EMV is a standard that defines the interaction between an integrated circuit (IC) cards and IC cards processing devices for payments. EMV stands for Europay, MasterCard and Visa, the companies that initiated the development of the EMV specifications in the mid 1990s. Over the years the initiator companies were joined by JCB, American Express and China Union Pay. Today the EMV standard is defined by the EMVCo LLC corporation.

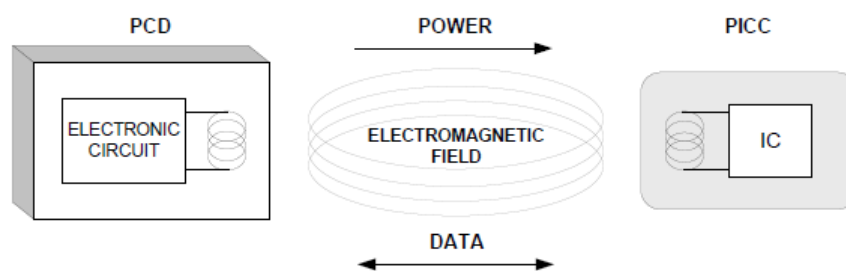
The EMV Contactless is based on ISO/IEC 14443 "Identification cards -- Contactless integrated circuit cards-- Proximity cards". It sets a standard for the usage of contactless systems for contactless payments.

In 2012 the EMVCo and NFC Forum agreed to work in collaboration on establishing a framework for the synchronization of the NFC Forum and EMVCo Specifications and the management of contactless product certification. The option R&S SMW-K89 enables you to generate signals in accordance with the NFC standard and the EMV Contactless standard thus allowing you to perform the tests needed to guarantee the proper performance of your devices.

The EMV specific abbreviations used in this manual as well as the different types of tag platforms/protocols are described in the EMV Contactless Specifications for Payment Systems. The specifications are available under [www.emvco.com](http://www.emvco.com).

### 4.1 Basics of data transmission with EMV contactless

A contactless system consists of two basic components: a contactless reader (PCD) and a transponder (PICC). The EMV contactless uses the electromagnetic near field of two conductor coils (a primary coil of the PCD and a secondary coil of the PICC) to couple the contactless reader and the transponder, see [Figure 4-1](#).



**Figure 4-1: PCD (contactless reader) and PICC (transponder) configuration**

The operating frequency is 13.56 MHz, and a bitrate of 106 kbit/s is used. Modulation schemes are amplitude shift keying (ASK) with different modulation depth (100 % or 10 %), amplitude on/off keying (OOK) and BPSK.

## 4.2 EMV contactless transmission technologies, modulation and coding

The EMV contactless has two main communication signal interfaces (based on ISO/IEC 14443):

- Type A
- Type B

The [Table 4-1](#) shows coding and modulation for the EMV Type A and the EMV Type B communication.

**Table 4-1: EMV contactless standards overview**

Standard	PCD-PICC / PICC-PCD	Coding	Modulation
Type A	PCD-PICC	Modified Miller	ASK 100%
	PICC-PCD	Manchester	Load modulation (OOK subcarrier)
Type B	PCD-PICC	NRZ-L	ASK 10%
	PICC-PCD	NRZ-L	Load modulation (BPSK subcarrier)

Refer to [Chapter 3.1.3, "Modulation scheme and coding"](#), on page 15 for a description of the used modulation schemes and coding.

## 5 NFC configuration and settings

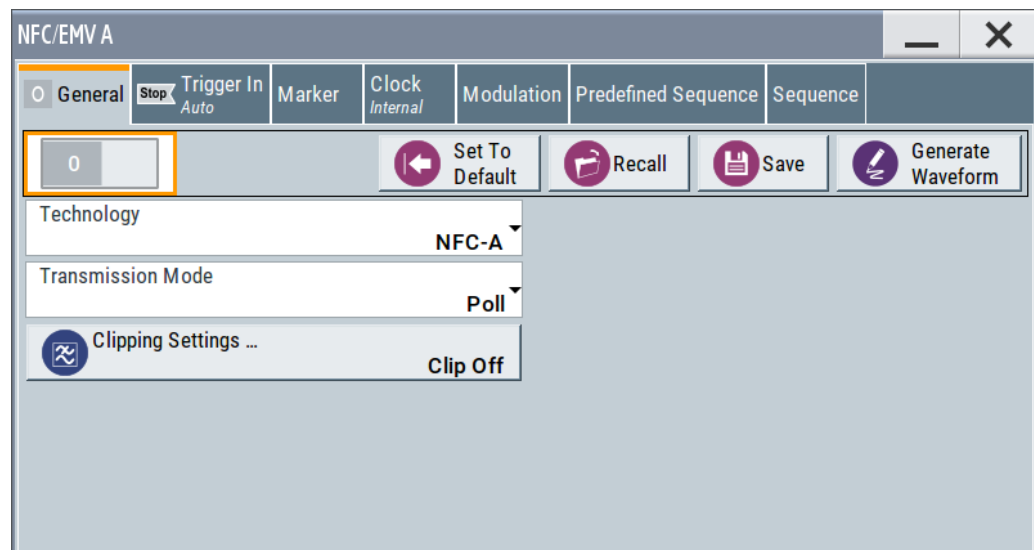
Access:

- Select "Baseband" > "NFC/EMV".

The remote commands required to define these settings are described in [Chapter 7](#), "Remote-control commands", on page 72.

### 5.1 General settings

This tab provides access to the default and the "Save/Recall" settings. The selected technology and transmission mode determine the available parameters.



State.....	24
Set to Default.....	25
Save/Recall .....	25
Data List Management.....	26
Generate Waveform File.....	26
Technology.....	26
Divisor (Bit Rate).....	26
Transmission Mode.....	26
Clipping Settings.....	27
Unmodulated Parts Voltage To Peak Voltage Ratio.....	27
Desired Voltage In Unmodulated Signal parts.....	27
Update Analog I/Q Settings For Desired Voltage.....	27

#### State

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



Remote command:

[ :SOURce<hw> ] :BB:NFC:STATe on page 77

### 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"
Technology	NFC A
Transmission mode	Poll
Clipping	Off
Command type (in "Sequence" dialog)	for NFC-A, Poll: SENS_REQ for NFC-B, Poll: SENSB_REQ for NFC-F, Poll: SENSF_REQ

Remote command:

[ :SOURce<hw> ] :BB:NFC:PRESet on page 77

### Save/Recall ...

Calls the "Save/Recall" dialog.

From this dialog the "Save/Recall Settings" windows for saving and recalling NFC configurations and the "File Manager" can be accessed.

NFC configurations are stored as files with the predefined file extension \*.nfc. Their file name and directory are user-definable.

The complete settings in the "NFC" dialog are saved and recalled.

"Recall NFC Setting"	Opens the "Recall Settings" window for loading a saved NFC configuration. The configuration of the selected (highlighted) file is loaded by pressing the "Select" button.
"Save NFC Setting"	Opens the "Save Settings" window for saving the current NFC signal configuration. The name of the file is specified in the file name entry field, the directory selected in the save into field. The file is saved by pressing the "Save" button.
"File Manager"	Calls the "File Manager". The "File Manager" is used to copy, delete and rename files and to create directories.

Remote command:

[ :SOURce<hw> ] :BB:NFC:SETTing:CATalog? on page 79

[ :SOURce<hw> ] :BB:NFC:SETTing:LOAD on page 79

[ :SOURce<hw> ] :BB:NFC:SETTing:STORe on page 79

[ :SOURce<hw> ] :BB:NFC:SETTing:DELeTe on page 79

**Data List Management...**

Calls the "Data List Management" dialog. This dialog is used to create and edit a data list.

All data lists are stored as files with the predefined file extension `*.dm_iqd`. Their file name and directory are user-definable.

The data lists must be selected as a data source from the subsection under the individual function.

**Note:** All data lists are generated and edited with the `SOURCE:BB:DM` subsystem commands. Files containing data lists usually end with `*.dm_iqd`. The data lists are selected as a data source for a specific function in the individual subsystems of the digital standard.

Remote command:

`[ :SOURCE<hw> ] :BB:NFC:CBLOCK<ch>:DATA` on page 95

`[ :SOURCE<hw> ] :BB:NFC:CBLOCK<ch>:DATA:DSELECTION` on page 95

**Generate Waveform File...**

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:NFC:WAVEFORM:CREATE` on page 78

**Technology**

Selects the NFC/EMV technology.

"NFC-A, NFC-B, NFC-F"

NFC technology. For details, see the NFC digital protocol technical specification.

"EMV Type A, EMV Type B"

EMV technology. For details, see the EMV technical specification.

Remote command:

`[ :SOURCE<hw> ] :BB:NFC:TECHNOLOGY` on page 77

**Divisor (Bit Rate)**

Available for "Technology > NFC-F", this parameter selects the used divisor (2 or 4) and determines the increased resulting bit rate of 212 Kbit/s or 424 Kbit/s respectively.

Remote command:

`[ :SOURCE<hw> ] :BB:NFC:DIVISOR` on page 76

**Transmission Mode**

Selects the transmission mode.

"Poll / Listen" Available for "Technology > NFC-A /NFC-B/ NFC-F"  
For details, see [Figure 3-1](#).

**"PICC to PCD / PCD to PICC"**

Available for "Technology > EMV Type A / EMV Type B"

For details, see [Chapter 3, "About the NFC digital standard"](#), on page 13 and [Chapter 4, "About the EMV contactless digital standard"](#), on page 22.

Remote command:

`[ :SOURce<hw> ] :BB:NFC:TMODe` on page 77

**Clipping Settings...**

Accesses the dialog to configure clipping, see [Chapter 5.10, "Clipping settings"](#), on page 67.

**Unmodulated Parts Voltage To Peak Voltage Ratio**

Available only for "Transmission Mode > Listen / PICC to PCD" and "State > On".

Displays the ratio of the voltage in the unmodulated parts of the signal to its peak value.

Remote command:

`[ :SOURce<hw> ] :BB:NFC:UPVoltage?` on page 78

**Desired Voltage In Unmodulated Signal parts**

Available only for "Transmission Mode > Listen / PICC to PCD" and "State > On".

Defines the desired voltage in unmodulated signal parts.

The displayed "Unmodulated ... Ratio" depends only on the signal and is not changed by the input of a "Desired Voltage".

See [Chapter 3.3, "Leveling aspects"](#), on page 20.

Remote command:

`[ :SOURce<hw> ] :BB:NFC:DVOLtage` on page 76

**Update Analog I/Q Settings For Desired Voltage**

Available only for "Transmission Mode > Listen / PICC to PCD" and "State > On".

Automatically adjusts the related parameters of the analog I and Q outputs to the desired voltage.

For a detailed description of all parameters, refer to section "I/Q analog output settings" in the manual of the R&S SMx.

See [Chapter 3.3, "Leveling aspects"](#), on page 20.

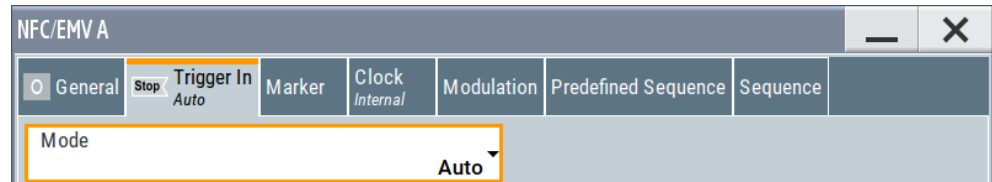
Remote command:

`[ :SOURce<hw> ] :BB:NFC:UAISetting` on page 77

## 5.2 Trigger settings

Access:

- Select "Baseband" > "NFC/EMV" > "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 5.5, "Local and global connectors settings"](#), on page 36.  
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 SMW 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 SMW user manual.


### Settings:

<a href="#">Trigger Settings Common to All Basebands</a> .....	29
<a href="#">Mode</a> .....	29
<a href="#">Signal Duration Unit</a> .....	29
<a href="#">Signal Duration</a> .....	30
<a href="#">Running/Stopped</a> .....	30
<a href="#">Time Based Trigger</a> .....	30
<a href="#">Trigger Time</a> .....	30

Arm.....	30
Execute Trigger.....	31
Source.....	31
Sync. Output to External Trigger/Sync. Output to Trigger.....	31
External Inhibit/Trigger Inhibit.....	32
External Delay/Trigger Delay.....	33

### Trigger Settings Common to All Basebands

To enable simultaneous signal generation in all basebands, the R&S SMW couples the trigger settings in the available basebands in any instrument's configuration involving signal routing with signal addition. For example, in MIMO configuration, routing and summing of basebands or of streams.

The icon  indicates that common trigger settings are applied.

You can access and configure the common trigger source and trigger mode settings in any of the basebands. An arm or a restart trigger event applies to all basebands, too. You can still apply different delay to each of the triggers individually.

### Mode

Selects trigger mode, i.e. determines the effect of a trigger event on the signal generation.

For more information, refer to chapter "Basics" in the R&S SMW user manual.

- "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:NFC\[:TRIGger\]:SEQUence](#) on page 119

### Signal Duration Unit

Available in "Single Trigger" mode. Defines the unit for describing the length of the signal sequence to be output.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:TRIGger:SLUNit](#) on page 119

**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:NFC:TRIGger:SLENgth](#) on page 118

**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:NFC:TRIGger:RMODe?](#) on page 118

**Time Based Trigger**

Requires trigger "Mode" > "Armed Auto"/"Single".

Activates time-based triggering with a fixed time reference.

The R&S SMW 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 SMW user manual.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:TRIGger:TIME\[:STATe\]](#) on page 121

**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 SMW. 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 SMW user manual.

"Date" Sets the date of the time-based trigger in format YYYY-MM-DD.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:TRIGger:TIME:DATE](#) on page 120

"Time" Sets the time of the time-based trigger in format hh:mm:ss.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:TRIGger:TIME:TIME](#) on page 120

**Arm**

Stops the signal generation until subsequent trigger event occurs.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:TRIGger:ARM:EXECute](#) on page 116

### Execute Trigger

Available only with internal trigger source and a trigger mode other than "Auto". Executes the trigger manually.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:TRIGger:EXECute](#) on page 116

### Source

The following sources of the trigger signal are available:

- "Internal"  
The trigger event is executed manually by the "Execute Trigger".
- "Internal (Baseband A/B)"  
The trigger event is provided by the trigger signal from the other basebands. If common trigger settings are applied, this trigger source is disabled.
- "External Global Trigger"  
The trigger event is the active edge of an external trigger signal provided and configured at the USER x connectors.
- "External Local Trigger"  
The trigger event is the active edge of an external trigger signal provided and configured at the local T/M/C connector.  
With coupled trigger settings, the signal has to be provided at the T/M/C1/2/3 connectors.
- "External Local Clock"  
The trigger event is the active edge of an external local clock signal provided and configured at the local T/M/C connector.  
With coupled trigger settings, the signal has to be provided at the T/M/C1 connector.
- "Baseband Sync In"  
Option: R&S SMW-B9  
In primary-secondary instrument mode, secondary instruments are triggered by the active edge of the synchronization signal.

"External Local Clock/Trigger" require R&S SMW-B10.

How to: ["Routing and activating a trigger signal"](#) on page 28

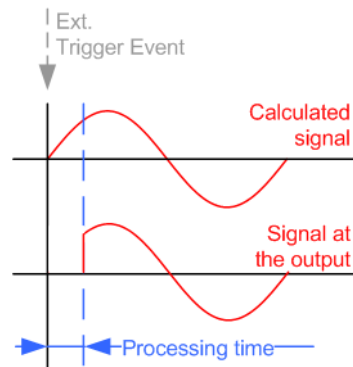
Remote command:

[\[:SOURce<hw>\]:BB:NFC:TRIGger:SOURce](#) on page 119

### Sync. Output to External Trigger/Sync. Output to Trigger

Enables signal output synchronous to the trigger event.

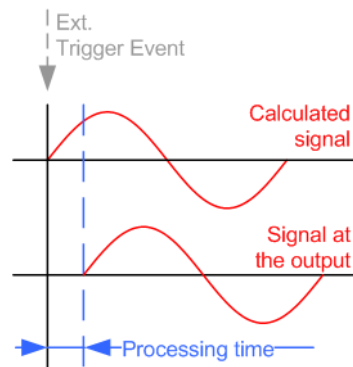
- "On"  
Corresponds to the default state of this parameter.  
The signal calculation starts simultaneously with the trigger event. Because of the processing time of the instrument, the first samples are cut off and no signal is output. After elapsing of the internal processing time, the output signal 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.



In primary-secondary instrument mode, this setting ensures that once achieved, synchronization is not lost if the baseband signal sampling rate changes.

Remote command:

`[ :SOURce<hw> ] :BB:NFC:TRIGger:EXTernal:SYNChronize:OUTPut`  
on page 117

### External Inhibit/Trigger Inhibit

Delays the trigger event of the signal from:

- The external trigger source
- The other path
- The other basebands (internal trigger), if common trigger settings are used.

Use this setting to:

- Synchronize the instrument with the device under test (DUT) or other external devices
- Postpone the signal generation start in the basebands compared to each other
- Compensate delays and align the signal generation start in multi-instrument setup

For more information, see chapter "Basics on ..." in the R&S SMW user manual.

Remote command:

`[ :SOURce<hw> ] :BB:NFC:TRIGger [ :EXTernal<ch> ] :DELay` on page 116  
`[ :SOURce<hw> ] :BB:NFC:TRIGger:OBASeband:DELay` on page 117



**External Delay/Trigger Delay**

Available on external triggering or on internal triggering via the second path.

Sets the duration for inhibiting a new trigger event subsequent to triggering.

In the "Retrigger" mode, every trigger signal causes signal generation to restart. This restart is inhibited for the specified duration.

For two-path instruments, the trigger inhibit can be set separately for each of the two paths.

Remote command:

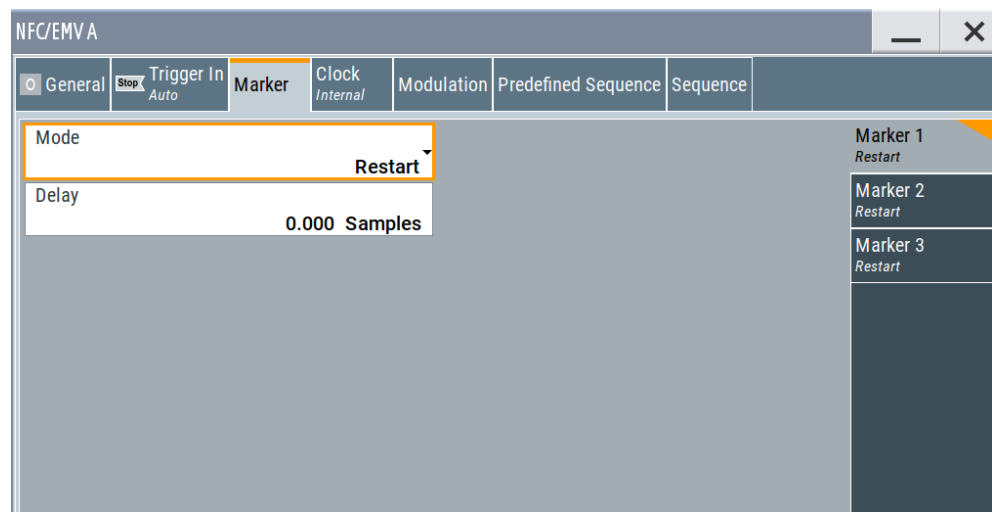
`[ :SOURce<hw> ] :BB:NFC:TRIGger [ :EXTernal<ch> ] :INHibit` on page 117

`[ :SOURce<hw> ] :BB:NFC:TRIGger:OBASeband:INHibit` on page 118

## 5.3 Marker settings

Access:

- Select "Baseband" > "NFC/EMV" > "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 5.5, "Local and global connectors settings"](#), on page 36.  
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 SMW adds the marker signal to the baseband signal. Also, R&S SMW 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 SMW user manual.

Settings:

Mode.....	34
Delay.....	35

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.

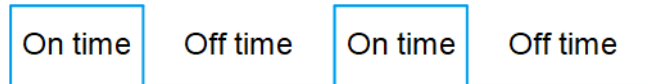
How to: ["Routing and activating a marker signal"](#) on page 33

- "Restart"            A marker signal is generated on every repetition of the complete frame sequence.
- "Pulse"             A regular marker signal is generated. The frequency is derived by dividing the sample rate by the divider. The input box for the divider opens when "Pulse" is selected, the resulting pulse frequency is displayed below it.
- "Pattern"           A marker signal that is defined by a bit pattern is generated. The pattern has a maximum length of 64 bits and is defined in an input field which opens when pattern is selected.

**"On/Off Ratio"**

Generated is a regular marker signal that is defined by an on/off ratio. A period lasts one on and off cycle.

The "ON Time" and "OFF Time" are each expressed as several samples.



Remote command:

`[ :SOURce<hw> ] :BB:NFC:TRIGger:OUTPut<ch>:MODE`

on page 121

`[ :SOURce<hw> ] :BB:NFC:TRIGger:OUTPut<ch>:PULSe:DIVider` on page 122

`[ :SOURce<hw> ] :BB:NFC:TRIGger:OUTPut<ch>:PULSe:FREQuency?` on page 123

`[ :SOURce<hw> ] :BB:NFC:TRIGger:OUTPut<ch>:PATtern` on page 122

`[ :SOURce<hw> ] :BB:NFC:TRIGger:OUTPut<ch>:ONTime` on page 122

`[ :SOURce<hw> ] :BB:NFC:TRIGger:OUTPut<ch>:OFFTime` on page 122

**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.

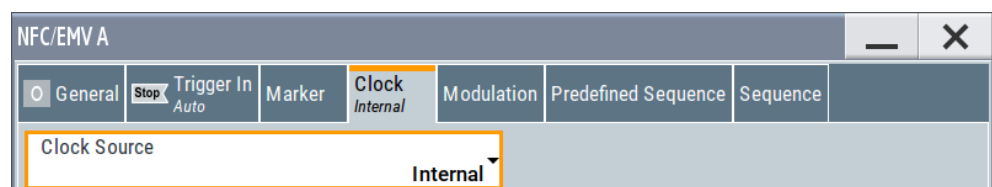
Remote command:

`[ :SOURce<hw> ] :BB:NFC:TRIGger:OUTPut<ch>:DELay` on page 123

## 5.4 Clock settings

Access:

- Select "Baseband" > "NFC/EMV" > "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.

- For external clock signals, define the connector for signal input. See [Chapter 5.5, "Local and global connectors settings"](#), on page 36.

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.

- Activate baseband signal generation. In the block diagram, set "Baseband" > "On".

The R&S SMW 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 SMW user manual.

#### Settings:

<a href="#">Clock Source</a> .....	36
<a href="#">Clock Mode</a> .....	36
<a href="#">Measured External Clock</a> .....	36

#### Clock Source

Selects the clock source.

- "Internal"  
The instrument uses its internal clock reference.
- "External Local Clock"  
Option: R&S SMW-B10  
The instrument expects an external clock reference at the local T/M/C connector.

How to: ["Defining the clock"](#) on page 35

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CLOCK:SOURce](#) on page 124

#### Clock Mode

Sets the type of externally supplied clock.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CLOCK:MODE](#) on page 123

#### Measured External Clock

Provided for permanent monitoring of the enabled and externally supplied clock signal.

Remote command:

[CLOCK:INPut:FREQuency?](#)

## 5.5 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.

## 5.6 Modulation settings

Access:

- Select "Baseband" > "NFC/EMV" > "Modulation".

The dialog provides common settings and slope settings to configure the signal modulation. Available parameters depend on the selected NFC technology and transmission mode. Some settings require active slope and RLC curve.

Settings:

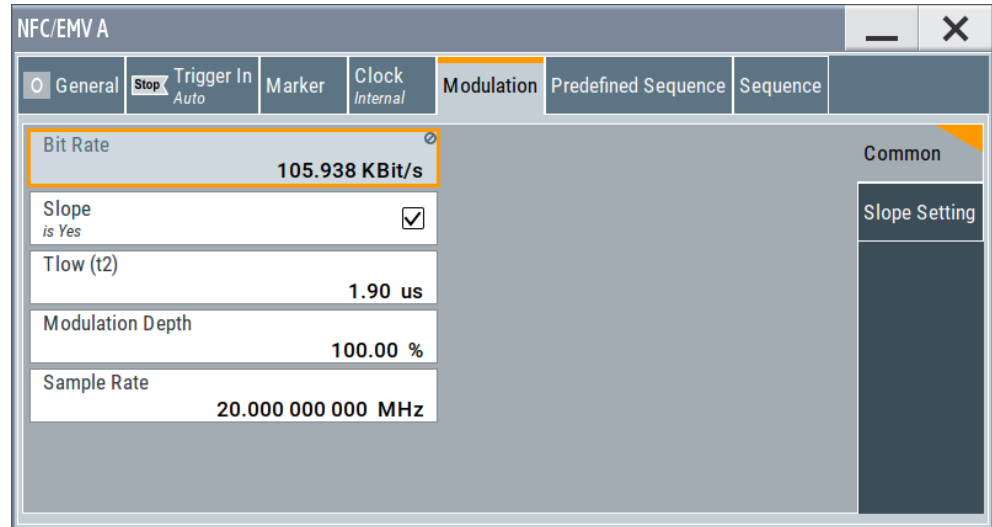
- [Common settings](#)..... 37
- [Slope settings](#)..... 41

### 5.6.1 Common settings

Access:

1. Select "NFC/EMV" > "Modulation".

2. Select "Common".



The dialog provides common settings to configure the signal modulation.

#### Settings:

Bit Rate.....	38
Modulation Index.....	38
Slope.....	39
Tlow (t2).....	39
Inverse Modulation.....	39
Modulation Depth.....	39
Baseband Output.....	40
Sample Rate.....	40
Slope Setting.....	40

#### Bit Rate

Indicates the current resulting bit rate in kbit/s.

Remote command:

[ :SOURce<hw> ] :BB:NFC:MSET:BRATE? on page 112

#### Modulation Index

Defines the signal's modulation index in %.

The modulation index represents the power drop during the low state transitions as a ratio of voltages at defined locations of the low state transition.

$$m_i = \frac{V_a - V_b}{V_a + V_b}$$

$V_a$  is the nominal high voltage and  $V_b$  is the nominal low voltage.

Remote command:

[ :SOURce<hw> ] :BB:NFC:MSET:MINDEX on page 113

**Slope**

Determines the transition between the modulated and unmodulated parts.



*Figure 5-1: Impact of the "Slope" parameter ("RLC Curve" = Off)*

"Off" A bursted signal with pulse like shape is generated. The transition time from high to low or low to high is only one sample.

"On" A longer transition time is used.

Remote command:

`[ :SOURce<hw> ] :BB:NFC:MSET:SLOPe` on page 112

**Tlow (t2)**

Available only for NFC-A in "Transmission Mode > Poll" and EMV A in "Transmission Mode > PCD to PICC".

Defines the signals low time (below 5%) in  $\mu\text{s}$ .

Remote command:

`[ :SOURce<hw> ] :BB:NFC:MSET:TLOW` on page 114

**Inverse Modulation**

Requires "NFC-B > Listen" and "NFC-F > Listen".

Activates inverse modulation.

Remote command:

`[ :SOURce<hw> ] :BB:NFC:MSET:IMODulation` on page 113

**Modulation Depth**

Requires NFC-A in "Transmission Mode > Poll" and EMV A in "Transmission Mode > PCD to PICC".

Sets the ASK modulation depth. The modulation depth indicates the magnitude of the voltage drop during the low state transition. The modulation depth is a percentage relative to the voltage of the carrier signal ( $V_1$ ).

Remote command:

`[ :SOURce<hw> ] :BB:NFC:MSET:MDEPth` on page 113

### Baseband Output

Requires "Listen" and "PICC to PCD" modes.

The default state is "On". When activated the signal at the baseband output changes between 0% and 100% voltage to be able to control the Reference Listeners. When deactivated baseband output delivers the envelope of the RF signal.

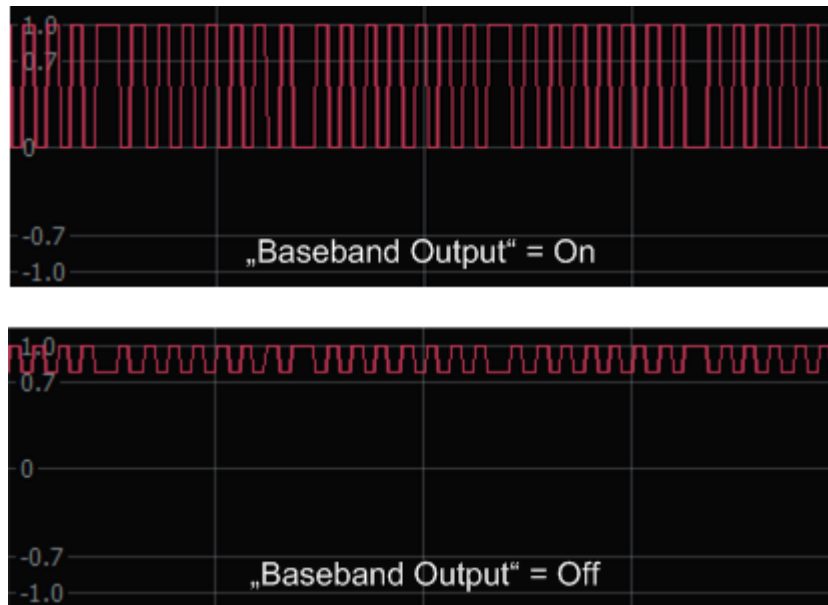


Figure 5-2: Impact of the parameter "Baseband Output"

Remote command:

`[ :SOURce<hw> ] :BB:NFC:MSET:BOUtpu` on page 112

### Sample Rate

In contrast to mobile radio standards (where this parameter is the "Sample Rate Variation"), the NFC standard does not prescribe a sample- or chiprate, but defines requirements, for example, for edge steepness.

At mobile radio standards, a change of the "Sample Rate Variation" does not change the number of samples per slot/frame/superframe etc., but rather plays the signal "faster" or "slower".

At NFC, the "Sample Rate" parameter changes the time resolution of signal generation, e.g. of how many samples an NFC-A bit duration is formed.

The 20 MSamples/s default value is a good trade-off between signal quality and required calculation time.

Remote command:

`[ :SOURce<hw> ] :BB:NFC:MSET:SRATe` on page 114

### Slope Setting

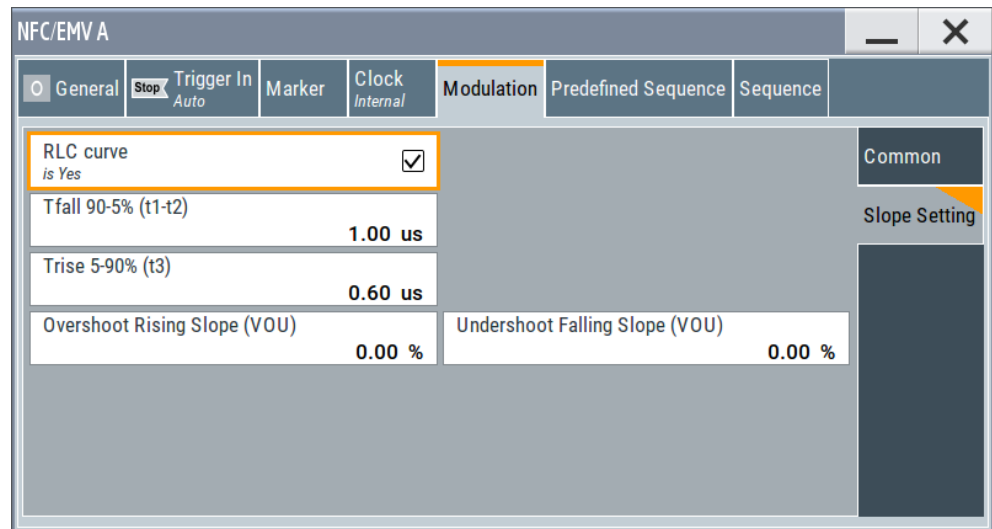
Access:



## 5.6.2 Slope settings

Access:

1. Select "NFC/EMV" > "Modulation".
2. To display the slope settings, select "Common" > "Slope" > "On".
3. Select "Slope Setting".



The dialog provides slope settings to configure the signal modulation.

### Settings:

RLC curve.....	41
Tfall 90-10%/Tfall 90-5% (t1-t2).....	42
Trise 10-90%/Trise 5-90% (t3).....	42
Overshoot Rising Slope (VOU).....	42
Undershoot Falling Slope (VOU).....	42

### RLC curve

Determines if an RLC curve (= discharge/charge curve of an RLC-circuit) is applied to the signal.

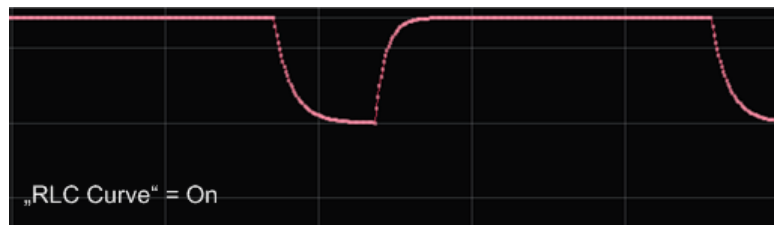


Figure 5-3: Impact of the "RLC Curve" parameter ("RLC Curve" = On)

"On"	An "RLC curve" is applied to the signal.
"Off"	A linear ramp is used.

Remote command:

[ :SOURce<hw> ] :BB:NFC:MSET:RCURve on page 114

#### **Tfall 90-10%/Tfall 90-5% (t1-t2)**

Defines the signals fall time (90 % to 5 % or 90% to 10%) in  $\mu\text{s}$ .

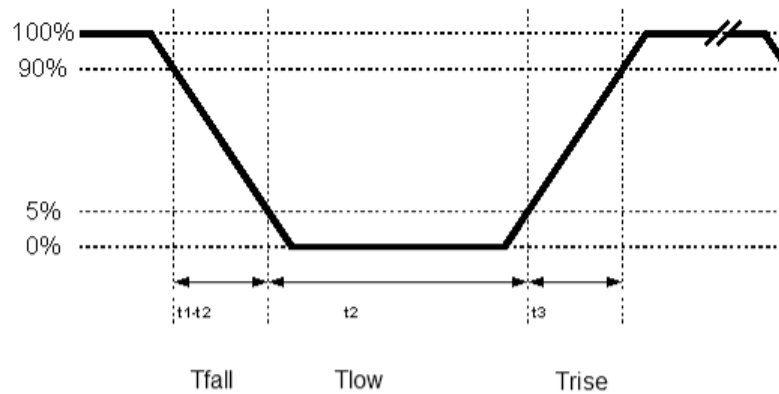


Figure 5-4: Tfall, Trise, Tlow at an NFC-A polling device to listening device

Remote command:

[ :SOURce<hw> ] :BB:NFC:MSET:TFAL1 on page 114

#### **Trise 10-90%/Trise 5-90% (t3)**

Defines the signals rise time (5 % to 90 % or 10 % to 90 %) in  $\mu\text{s}$ , see also "Tfall 90-10%/Tfall 90-5% (t1-t2)" on page 42.

Remote command:

[ :SOURce<hw> ] :BB:NFC:MSET:TRISE on page 114

#### **Overshoot Rising Slope (VOU)**

Determines the size of the overshoot after the rising slope. The parameter corresponds to the value  $V_{OU}$  in the NFC Analog Technical Specification. Overshoot Rising Slope is in percent of the difference between the nominal high voltage to the nominal low voltage, according to the following formula:

$$\text{Overshoot in Volts} = V_{OU} \times (V_a - V_b),$$

$V_a$  is the nominal high voltage and  $V_b$  is the nominal low voltage.

Remote command:

[ :SOURce<hw> ] :BB:NFC:MSET:OSRise on page 113

#### **Undershoot Falling Slope (VOU)**

Determines the size of the undershoot (ringing) after the falling slope. The parameter corresponds to the value  $V_{OU}$  in the NFC Analog Technical Specification. Undershoot Falling Slope is in percent of the difference between the nominal high voltage to the nominal low voltage, according to the following formula:

$$\text{Undershoot in Volts} = V_{OU} \times (V_a - V_b),$$

$V_a$  is the nominal high voltage and  $V_b$  is the nominal low voltage.

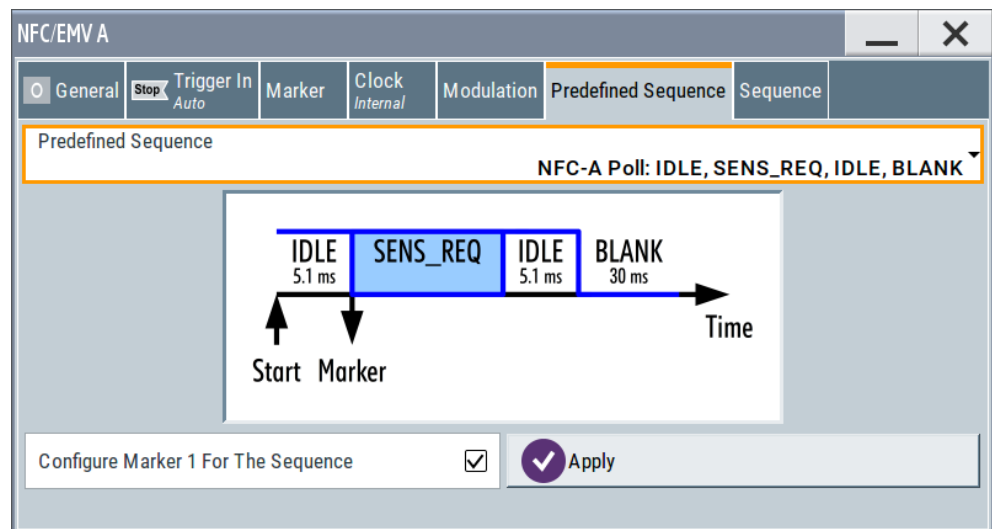
Remote command:

[ :SOURce<hw> ] :BB:NFC:MSET:USFall on page 115

## 5.7 Predefined sequence

This tab provides access to the sequence configuration.

- To access the predefined sequence tab, select "Baseband > NFC/EMV > Predefined Sequence".



This dialog contains the parameters to define a predefined sequence for transmission modes "Poll" and "PCD to PICC".

### Predefined Sequence

Selects a predefined sequence.

Remote command:

[ :SOURce<hw> ] :BB:NFC:PRED:SEquence on page 80

### Configure Marker 1 For The Sequence

**Note:** Available for signal generators only.

Enables Marker 1 as shown in the picture of the dialog.

Remote command:

[ :SOURce<hw> ] :BB:NFC:PRED:CNFMarker on page 80

### Apply

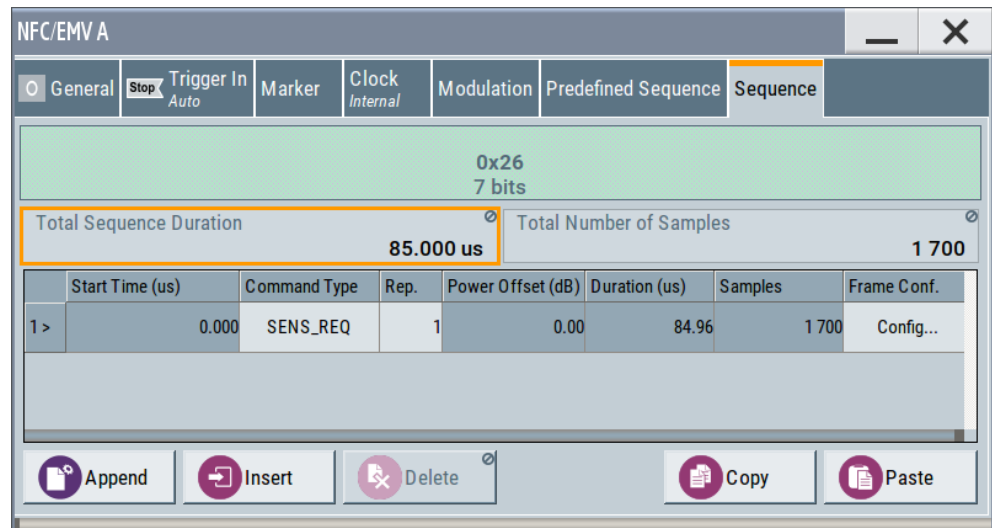
Activates the selected "Predefined Sequence" and marker status.

Remote command:

[ :SOURce<hw> ] :BB:NFC:PRED:APPLY on page 80

## 5.8 Sequence configuration settings

- To access the sequence configuration settings, select "Baseband > NFC/EMV > Sequence".



In this dialog you can define elements of the frame sequence, (e.g. start time, repetition, frame duration, power offset) for each command block.

The example in the screenshot shows some available command types for "Technology > NFC-A", "Transmission Mode > Poll".

For each command block selected, the resulting frame content of the frames of the command block appears on top of the dialog. The content depends not only on the selected command type, but for some command types also on the parameters set in the "Frame Configuration" dialog.

For available command types see [Table 5-1](#).

### Total Sequence Duration

Displays the overall sequence duration.

$$\text{Total Sequence Duration} = \text{Total Number of Samples} / \text{Sample Rate}$$

For an example, see [Figure 3-10](#).

Remote command:

`[ :SOURCE<hw> ] :BB:NFC:SCONfiguration:TSDuration?` on page 84

### Total Number of Samples

Displays the total number of samples allocated to the current sequence configuration. The displayed value is the sum of the samples of the individual commands.

$$\text{Total Number of Samples} = \text{Samples}_1 + \dots + \text{Samples}_N$$

For an example, see [Figure 3-10](#).

Remote command:

`[ :SOURce<hw> ] :BB:NFC:SCONfiguration:TNSamples?` on page 84

### Sequence Table

Contains the elements of the command sequence.

The first table column shows the successive command block number.

### Start Time ← Sequence Table

Displays the exact start time of the corresponding command (in  $\mu\text{s}$ ). The value is calculated as the sum of the samples of all preceding commands converted to time.

$$\text{Start Time}_N = (\text{Samples}_1 + \dots + \text{Samples}_{N-1}) / \text{Sample Rate}$$

For an example, see [Figure 3-10](#).

Remote command:

`[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:STIME?` on page 84

### Command Type ← Sequence Table

Determines the command type for each command block.

For the different technologies and transmission modes you can select command types as listed in [Table 5-1](#).

The different types of tag platforms (used in the table header) are described in the NFC Digital Protocol Technical Specification.

**Table 5-1: Available command types for "Technology > NFC-A" and the corresponding SCPI-command parameters (in brackets).**

"Transmission Mode"	Platform / Protocol				
	General command type	Type 1 Tag	Type 2 Tag	Type 4A Tag	NFC-DEP
<b>"Poll"</b>	"ALL_REQ" (ALAQ)	"RID" (RDAQ)	"READ" (T2RQ)	"RATS" (RATQ)	"ATR_REQ" (ATRQ)
	"SENS_REQ" (SNAQ)	"RALL" (RLAQ)	"WRITE" (T2WQ)	"DATA" in "ISO-DEP" (T4AD)	"PSL_REQ" (PSLQ)
	"SDD_REQ" (SDAQ)	"READ"(T1RQ)	"SECTOR SELECT" (SSLQ)		"DEP_REQ" (DEPQ)
	"SEL_REQ" (SLAQ)	"WRITE-E" (WREQ)			"DSL_REQ" (DSLQ)
	"SLP_REQ" (SPAQ)	"WRITE-NE" (WNEQ)			"RLS_REQ" (RLSQ)
	"GENERIC" (GENE)	"RSEG" (RSGQ)			
	"BLANK" (BLNK)	"READ8" (RD8Q)			
	"IDLE" (IDLE)	"WRITE-E8" (WE8Q)			
		"WRITE-NE8" (WN8Q)			
<b>"Listen"</b>	"SENS_RES" (SNAS)	"RID" (RDAS)	"READ" (T2RS)	"ATS" (ATSS)	"ATR_RES" (ATRS)
	"SDD_RES" (SDAS)	"RALL" (RLAS)	"ACK" (ACK)	"DATA" in "ISO-DEP" (T4AD)	"PSL_RES" (PSLS)
	"SEL_RES" (SLAS)	"READ" (T1RS)	"NACK" (NACK)		"DEP_RES" (DEPS)
	"BLANK" (BLNK)	"WRITE-E" (WRES)			"DSL_RES" (DSLS)
	"IDLE" (IDLE)	"WRITE-NE" (WNES)			"RLS_RES" (RLSS)
		"RSEG" (RSGS)			
		"READ8" (RD8S)			
		"WRITE-E8" (WE8S)			
		"WRITE-NE8" (WN8S)			

**Table 5-2: Available command types for "Technology > NFC-B" and corresponding SCPI-command parameters (in brackets).**

"Transmission Mode"	General command type	Platform / Protocol
		Type 4 B Tag
"Poll"	"ALLB_REQ" (ALBQ) "SENSB_REQ" (SNBQ) "SLOT_MARKER" (SMAR) "SLPB_REQ" (SPBQ) "GENERIC" (GENE) "BLANK" (BLNK) "IDLE" (IDLE)	"ATTRIB" (ATBQ) "DATA" in "ISO-DEP" (T4BD)
"Listen"	"SENSB_RES" (SNBS) "SLPB_RES" (SPBS) "BLANK" (BLNK) "IDLE" (IDLE)	"ATTRIB" (ATBS) "DATA" in "ISO-DEP" (T4BD)

**Table 5-3: Available command types for "Technology > NFC-F" and corresponding SCPI-command parameters (in brackets).**

"Transmission Mode"	General command type	Platform / Protocol	
		Type 3 Tag	NFC-DEP
"Poll"	"SENSF_REQ" (SNFQ) "GENERIC" (GENE) "BLANK" (BLNK) "IDLE" (IDLE)	"CHECK" (CHKQ) "UPDATE" (UPDQ)	"ATR_REQ" (ATRQ) "PSL_REQ" (PSLQ) "DEP_REQ" (DEPQ) "DSL_REQ" (DSLQ) "RLS_REQ" (RLSQ)
"Listen"	"SENSF_RES" (SNFS) "BLANK" (BLNK) "IDLE" (IDLE)	"CHECK" (CHKS) "UPDATE" (UPDS)	"ATR_RES" (ATRS) "PSL_RES" (PSLS) "DEP_RES" (DEPS) "DSL_RES" (DSLRS) "RLS_RES" (RLSS)

**Table 5-4: Available command types for "Technology >EMV Type A" and corresponding SCPI-command parameters (in brackets).**

"Transmission Mode"	General command type
"PCD to PICC"	"WUPA" (ALAQ) "REQA" (SNAQ) "ANTICOLLISION" (SDAQ) "SELECT" (SLAQ) "HLTA" (SPAQ) "RATS" (RATQ) "DATA_Type_A" (T4AD) "BLANK" (BLNK) "IDLE" (IDLE)
"PICC to PCD"	"ATQA" (SNAS) "ANTICOLLISION" (SDAS) "SAK" (SLAS) "ATS" (ATSS) "DATA_Type_A" (T4AD) "BLANK" (BLNK) "IDLE" (IDLE)

**Table 5-5: Available command types for "Technology >EMV Type B" and corresponding SCPI-command parameters (in brackets).**

"Transmission Mode"	General command type
"PCD to PICC"	"WUPB" (ALBQ) "REQB" (SNBQ) "HLTB" (SPBQ) "ATTRIB" (ATBQ) "DATA_Type_B" (T4BD) "BLANK" (BLNK) "IDLE" (IDLE)
"PICC to PCD"	"ATQB" (SNBS) "HLTB" (SPBS) "ATTRIB" (ATBS) "DATA_Type_B" (T4BD) "BLANK" (BLNK) "IDLE" (IDLE)

**Note:** The IDLE command produces an unmodulated signal part of a configurable length while the BLANK command produces a signal part without any output signal.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:CTYPE on page 82



**Rep. ← Sequence Table**

Determines the number of times to repeat the generation of a frame.

See also [Figure 3-10](#).

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:REPetition](#) on page 83

**Power Offset (dB) ← Sequence Table**

Determines the value of the power offset in dB.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:POFFset](#) on page 83

**Duration (μs) ← Sequence Table**

For "Command Type > BLANK/IDLE", determines the frame period in μs. For all other commands, the duration is displayed as defined in the standard.

For an example, see [Figure 3-10](#).

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:DURation](#) on page 82

**Samples ← Sequence Table**

Displays the number of the samples used for the command. For the calculation of the value, the [Duration \(μs\)](#) is converted to samples and rounded up.

Samples = ceiling (Duration \* Sample Rate)

For an example, see [Figure 3-10](#).

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:SAMPles?](#) on page 83

**Frame Configuration ← Sequence Table**

Accesses the "Frame Configuration " dialog for each command block, see [Chapter 5.9](#), "[Frame configuration settings](#)", on page 50.

**Append, Insert, Delete, Copy, Paste**

General functions for editing the sequence configuration, as append, insert, delete, copy or paste a command block.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CBLOCK:APPend](#) on page 81

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:INSert](#) on page 82

[\[:SOURce<hw>\]:BB:NFC:ICBLOCK](#) on page 82

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:DELete](#) on page 82

[\[:SOURce<hw>\]:BB:NFC:DCBLOCK](#) on page 82

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:COPY](#) on page 81

[\[:SOURce<hw>\]:BB:NFC:CCBLOCK](#) on page 81

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:PASTE](#) on page 83

[\[:SOURce<hw>\]:BB:NFC:PCBLOCK](#) on page 83

## 5.9 Frame configuration settings

For each "Command Type" listed in the [Table 5-1](#) a "Frame Configuration" dialog is available.

1. Select "General > Technology > NFC-B".
2. Select "General > Transmission Mode > Listen".
3. To access the frame configuration dialog, select "Config..." in the corresponding row in the "Sequence Configuration" table.

NFC/EMV: SENSB_RES Frame Configuration 1			
0x50 1 byte	NFCID0 0x01234567 4 bytes	Application Data 0x01234567 4 byte	Protocol Info 0x772140 3 bytes
NFCID0 (hex) 0123 4567		App. Data Coding Proprietary	Common
FSC 32	Minimum TR2 1792/fc		App. Data Set.
FWI 4	Advanced Protocol Feature <i>is Not Supported</i>		Direction Set.
DID <i>is Not Supported</i>	Extended SENSB_RES <i>is Not Supported</i>		

The top of the "Frame Configuration" dialog shows the resulting frame content for the current settings. The parameters and functions in the "Frame Configuration" dialog depend on the "Command Type" selected in the sequence configuration.

For the following command types only the bit value of the frame is displayed because the value is fixed:

ALL\_REQ, ACK, SENS\_REQ, SLP\_REQ, RID

### 5.9.1 Parameters of the frame configuration dialog

In the following chapter the settable parameters of all command types are listed alphabetically. Each command type uses only some of these parameters. The availability of some settings depends on other settings in the same dialog and on the selected "Technology" and "Transmission Mode".



Some parameters have the same functionalities for the NFC and the EMV technologies, but are named differently for the specific technology. In the following chapter the equivalent parameters are described only once. The parameter names for both technologies are contained in the parameter title and both names are divided by a "/". The parameter name for the NFC technology is written on the first place.

**Example:**

"NFCID1 (hex) / UID (hex)", where "NFCID1 (hex)" is the parameter name for the NFC technology and "UID (hex)" the parameter name for the EMV technology.

## ACK, NACK

Available only for "PDU Type > ACK-NACK" or "Block Type > R-block".

Selects ACK or NACK.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBlock<ch>:ANSelection on page 88

### Advanced Protocol Features supported

Enables/disables the support of advanced protocol features.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:APFSupported on page 88

### App. Data Set.

Access:

Select "General > Technology > NFC-B".

Select "General > Transmission Mode > Listen".

Select the "Sequence" tab.

For "Command Type > SENSB\_RES", select "Frame Conf. > Config... > App. Data Set.".

NFC/EMV: SENS_B_RES Frame Configuration 1			
0x50 1 byte	NFCID0 0x01234567 4 bytes	Application Data 0x00 + CRC_B(AID) + 0x11 4 byte	Protocol Info 0x772144 3 bytes
No. of Applications 1		Tot No. Apps in the PICC 1	
AID Length 1 bytes		AID (hex) 00...	
AFI 0			
		Common	
		App. Data Set.	
		Direction Set.	

**Application Data ← App. Data Set.**

If "Application Data Coding > Proprietary" is used, enters the application data in hex format.

Remote command:

[ :SOURCE<hw> ] :BB:NFC:CBLOCK<ch>:ADATa on page 87

**AID Length ← App. Data Set.**

For "Application Data Coding > CRC-B", determines the length of AID.

Remote command:

[ :SOURCE<hw> ] :BB:NFC:CBLOCK<ch>:ALENgt h on page 88

**AID (hex) ← App. Data Set.**

Determines the value of AID.

Remote command:

[ :SOURCE<hw> ] :BB:NFC:CBLOCK<ch>:AID on page 88

**AFI ← App. Data Set.**

Sets the application family being selected.

Remote command:

[ :SOURCE<hw> ] :BB:NFC:CBLOCK<ch>:AFI on page 88

**Application Data Coding**

Determines the way the application data is coded: with a "Proprietary" code or using a "CRC-B" compressing method.

Remote command:

[ :SOURCE<hw> ] :BB:NFC:CBLOCK<ch>:ADCoding on page 87

**ATN or Timeout**

Available only for "PDU Type > Supervisory".

Determines whether a "ATN" (Attention) or "Timeout" supervisory PDU type is used.

Remote command:

[ :SOURCE<hw> ] :BB:NFC:CBLOCK<ch>:ATIMEout on page 91

**BCC Error**

Used for error detection. If enabled, an error is added intentionally to the BCC (Block Check Character) by adding 1 to the BCC Byte.

Remote command:

[ :SOURCE<hw> ] :BB:NFC:CBLOCK<ch>:BCCError on page 91

**Bit Frame SDD / Bit Frame Anticollision**

Determines the Bit frame SDD / Anticollision.

Remote command:

[ :SOURCE<hw> ] :BB:NFC:CBLOCK<ch>:BFSDD on page 92

[ :SOURCE<hw> ] :BB:NFC:CBLOCK<ch>:BFANTicollision on page 91

**Block List, Block Data, Block List Configuration**

Available with "Command Type > CHECK". Accesses the "Block List Configuration" dialog.

The available functions in the "Block List Configuration" dialog depend on the selected "Transmission Mode".

With "Transmission Mode > Poll" the block list appears.

NFC/EMV: Block List Configuration			
	Length	Service Code List Order	Block No.
1	2 bytes	0	0

The number of rows in the block list is determined by the parameter [Number of Blocks](#).  
With "Transmission Mode > Listen" the block data appears.

NFC/EMV A: Block Data Configuration	
	Block Data
1	FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF

- "Length" Sets the block length in bytes.  
Remote command:  
`[ :SOURCE<hw> ] :BB:NFC:CBLOCK<ch>:BLOCK<st>:LEN`  
on page 92
- "Service Code List Order" Sets the service code list order.  
Remote command:  
`[ :SOURCE<hw> ] :BB:NFC:CBLOCK<ch>:BLOCK<st>:SLORDER`  
on page 93

"Block Number"	<p>Sets the block number.</p> <p>Remote command:  <a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:BLOCK&lt;st&gt;:BNUMBER</a>  on page 92</p>
"Block Data"	<p>Enters the block data in hex format.</p> <p>Remote command:  <a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:BLOCK&lt;st&gt;:BDATA</a>  on page 92</p>

**Block Number (BNo)**

Selects the block number to be read or written.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:BNO](#) on page 93

**Block Number (for I-block type or R-block type)**

Available for I-block type or R-block type. Indicates if a valid I-block or a valid R (ACK) block is received.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:IBNUMBER](#) on page 100

**Block or Byte Selection (ADD)**

Selects a block/byte to be read or written.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:BLKSelection](#) on page 92

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:BYTSelection](#) on page 94

**Block Type**

Determines the used block type.

"I-block"	Used to convey information for use by the application layer.
"R-block"	Used to convey positive or negative acknowledgements. An R-block never contains an INF field. The acknowledgement relates to the last received block.
"S-block"	Used to exchange control information between the reader/writer and the card emulator.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:BTYPe](#) on page 93

**Chaining**

Available only for "Block Type > I-block".

Determines if chaining is applied.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:CHAINing](#) on page 94

**Configuration Type**

Only used in case NFCID1 is not completed.(Cascade bit == "On").

Determines what platform or protocol the device in Listen mode is configured for.

- "Type 2 Tag" This platform uses the following characteristics of NFC-A:
- synchronization mechanism
  - bit level coding
  - transmits Commands and Responses in NFC-A standard frames, except for the ACK and NACK Response.
- "Type 4A Tag" This platform uses the following characteristics of NFC-A:
- synchronization mechanism
  - bit level coding
  - transmits commands and responses in NFC-A standard frame format.
- "NFC-DEP" This protocol uses the following characteristics of NFC-A or NFC-F, depends on the configuration :
- sequence format
  - bit level coding
  - frame format.
- "NFC-DEP&Type 4A Tag"
- Used for devices capable of both, NFC-DEP and Type 4A Tag.  
For more details see the NFC Digital Protocol Technical Specification.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:CFGType](#) on page 94

### Data (hex)

Sets the data for the corresponding frame.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:APGeneric:BOData](#) on page 89

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:APGeneric:SHData](#) on page 89

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:APGeneric:STData](#) on page 90

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:BPGeneric:DATA](#) on page 90

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:FPGeneric:DATA](#) on page 90

### Data Length

Determines the length of the transmitted user data / general data.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:DATA:LENGTH](#) on page 95

### Data Source (Data)

Determines the data source type for the frame.

"All 0 / All 1" Generates 0 or 1 data.

"PN9 / 11 / 15 / 16 / 20 / 21 / 23"

Generates PRBS data in accordance with ITU-T with period lengths between  $2^9-1$  and  $2^{23}-1$ .

"Data List" Uses data from a programmable data list. The data can be generated with the binary editor in the instrument or externally with any editor. Data list files are selected from the "Select Data List" dialog.

"Pattern" Defines a bit pattern. For Pattern input select "Data Pattern".

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:DATA on page 95

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:DATA:DSELECTION on page 95

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:DATA:PATTERN on page 95

### DESELECT or WTX

Available only for "Block Type > S-block".

Determines whether a "DESELECT" or a "WTX" (waiting) is sent.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:DWSELECTION on page 99

### DID Supported / CID Supported

Determines if DID / CID is supported.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:DSUPPORTED on page 99

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:CSUPPORTED on page 95

### DID (DID field) / CID (CID field)

Determines the value of DID (Device Identification Number) / CID (Cryptogram Information Data).

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:DID on page 96

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:CID on page 94

### DID following (I-block type, R-block type or S-block type)

Determines if a DID is following.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:DFOLLOWING on page 96

### Direction Set.

Select "General > Technology > NFC-B".

Select "General > Transmission Mode > Listen".

Select the "Sequence" tab.

For "Command Type > SENSB\_RES", select "Frame Conf. > Config... > Direction Set."



NFC/EMV: SENSB_RES Frame Configuration 1			
0x50 1 byte	NFCID0 0x01234567 4 bytes	Application Data 0x01234567 4 bytes	Protocol Info 0x77614900 4 bytes
D(LISTEN->POLL)=D(POLL->LISTEN) <input type="checkbox"/>			Common
D(L->P)=8 <i>is Supported</i> <input checked="" type="checkbox"/>		D(L->P)=4 <i>is Supported</i> <input checked="" type="checkbox"/>	App. Data Set.
D(L->P)=2 <i>is Supported</i> <input checked="" type="checkbox"/>			Direction Set.
D(P->L)=8 <i>is Supported</i> <input checked="" type="checkbox"/>		D(P->L)=4 <i>is Supported</i> <input checked="" type="checkbox"/>	
D(P->L)=2 <i>is Supported</i> <input checked="" type="checkbox"/>			

**D(LISTEN->POLL)=D(POLL->LISTEN) / D(PICC->PCD)=D(PCD->PICC) ← Direction Set.**

Determines if the same bit rate divisor for both directions is supported.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:DEQD on page 96

**D(L->P=8), D(L->P=4), D(L->P=2) ← Direction Set.**

In the transmission direction listen to poll, indicate support of the corresponding divisor, i.e. determine the bit rate capability.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:DLP8 on page 97

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:DLP4 on page 97

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:DLP2 on page 96

**D(P->L=8), D(P->L=4), D(P->L=2) ← Direction Set.**

In the transmission direction poll to listen, indicate support of the corresponding divisor, i.e. determine the bit rate capability.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:DPL8 on page 98

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:DPL4 on page 97

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:DPL2 on page 97

**Divisor (LISTEN to POLL), Divisor (POLL to LISTEN) / Divisor (PCD to PICC),  
Divisor (PICC to PCD)**

Available for "Command Type > ATTRIB".

Set the divisor in the corresponding transmission direction.

Remote command:

[ :SOURce<hw>] :BB:NFC:CBLock<ch>:DLTPoll on page 97

[ :SOURCE<hw> ] :BB:NFC:CBLOCK<ch>:DPTListen on page 98

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:DPPicc on page 98

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:DPPCd on page 98

### DRI

Selects the divisor (1, 2, 4, 6, 8, 16, 32, 64) in communication direction from target to initiator. The divisor determines the bit rate.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:DRI on page 98

### DSI

Selects the divisor (1, 2, 4, 6, 8, 16, 32, 64) in communication direction from initiator to target. The divisor determines the bit rate.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:DSI on page 99

### EoD (CRC)

Selects if the EoD is present or not. The EoD contains a 2-byte CRC.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:APGeneric:STEPresent on page 91

### Extended SENSB\_RES / Extended ATQB

Determines if "Extended SENSB\_RES" / "Extended ATQB" is supported.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:ESSupported on page 99

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:EASupported on page 99

### Frame Type

Selects a frame type for "Command Type > GENERIC".

"Short Frame"	Used to initiate communication. A short frame consists of an SoF, up to 7 data bits and an EoF.
"Standard Frame"	Used for data exchange. A standard frame consists of an SoF, n*(8 data bits + odd parity bit ) where n≥1 and for the case of a Poll - Listen communication an EoF.
"Bit Oriented SDD Frame"	Used for collision resolution. A bit oriented SDD frame results from a standard frame of 7 bytes that is divided into two parts.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:APGeneric:FTYPE on page 89

### FSC

Selects the maximum frame size in bytes.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:FSC on page 99

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:MFSize on page 101

**FWI**

Determines the FWI (Frame Waiting time Integer) which is needed to calculate the FWT (Frame Waiting Time).

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:FWI on page 100

**General Data**

Determines if the bytes with General Data are available.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:GDAvailable on page 100

**Global Block Selection (ADD)**

Selects 8-byte block to be read or written.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:GBSelection on page 100

**k**

Determines the number of historical bytes to be used. To set the bytes  $T_1$  to  $T_k$  themselves, use the parameter **T1 to Tk**.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:KParameter on page 101

**Length Reduction**

According to the NFC specification, the length reduction bits (LR) are used to restrict the payload size.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:LREduction on page 101

**Lock Control or Status**

Enables/disables status information on lock for the corresponding block ("BLOCK-1" to "BLOCK-C").

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:BLOCK<st>:LOCKed on page 93

**MBLI**

Determines the Maximum Buffer Length Index (MBLI).

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:MBLI on page 101

**MI (more information) Chaining**

If enabled, the More Information (MI) bit indicates chaining. Chaining indicates that a larger data block is split into several PDUs and the current PDU contains only a part of the data.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:MIChaining on page 101

**Minimum TR0, TR1, TR2**

Indicates the minimum value of TR0/TR1/TR2 supported. The fc stands for the carrier frequency.

"1008/fc, 768/fc, 256/fc"

Minimum supported TR0

"1254/fc, 1024/fc, 256/fc"

Minimum supported TR1

"1792/fc, 3328/fc, 5376/fc, 9472/fc"

Minimum supported TR2.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:MTR0 on page 102

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:MTR1 on page 102

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:MTR2 on page 102

**NACK**

Determines the value of NACK.

"0 / 1 / 4 / 5" Value of NACK in hex.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:NACK on page 102

**NAD**

Available only for "NAD following > On".

Determines the value of NAD.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:NAD on page 103

**NAD following**

Determines if NAD is following.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:NFOLlowing on page 103

**NAD Supported**

Enables/disables the support of NAD.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:NSUPported on page 105

**NFCID0 (hex) / PUPI (hex)**

Determines the entire value of NFCID0/ PUPI. The length of NFCID0 /PUPI is fixed to 4 in octet.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:NID0 on page 103

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:PUPI on page 106

**NFCID1 (hex) / UID (hex)**

Determines the entire value of NFCID1/ UID.

The length of NFCID1/ UID is configurable to up to 10 bytes.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:NID1 on page 103

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:UID on page 111

#### **NFCID1 not complete / UID not complete**

Determines whether NFCID1 / UID is complete or not.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:NNComplete on page 104

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:UNComplete on page 111

#### **NFCID1 Size / UID Size**

Determines the size of NFCID1/ UID.

"Single "            The size is 4 bytes.

"Double"            The size is 7 bytes.

"Triple"            The size is 10 bytes.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:NSIZE on page 104

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:USIZE on page 111

#### **NFCID2 Format Type**

Indicates if the NFCID2 format is for NFC-DEP Protocol or Tag Type 3 platform.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:N2FType on page 102

#### **NFCID2 (hex)**

Determines the entire value of NFCID2. The value of Byte 2 in NFCID2 is fixed to "FE", except at command type 3.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:NID2 on page 103

#### **Number of Applications**

Determines the number of applications.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:NOApplications on page 104

#### **Number Of Bits**

Sets the length of a short frame.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:APGeneric:SHLength on page 90

#### **Number Of Bits For Part 1**

Sets the length of the first part of a bit oriented SDD frame.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:APGeneric:BOLength on page 89

**Number of Blocks**

Determines the number of blocks.

Select the [Block List](#), [Block Data](#), [Block List Configuration](#) to access the dialog with further settings.

Remote command:

[\[:SOURCE<hw>\]:BB:NFC:CBlock<ch>:NBLOCKS](#) on page 103

**Number Of Data Bytes**

Shows the total length of a frame in bytes.

The length for the different NFC technologies is calculated as follows:

- NFC A: the sum of the "Number Of Payload Bytes" and if "EoD (CRC)" is present, 2 additional bytes are added
- NFC B: the sum of the "Number Of Payload Bytes" and 2 additional bytes added for "EoD (CRC)"
- NFC F: the sum of the "Number Of Payload Bytes", 2 additional bytes added for "EoD (CRC)" and 1 additional byte added for SoD

Remote command:

[\[:SOURCE<hw>\]:BB:NFC:CBlock<ch>:APGeneric:STDLength?](#) on page 90

[\[:SOURCE<hw>\]:BB:NFC:CBlock<ch>:BPGeneric:DLENGth?](#) on page 90

[\[:SOURCE<hw>\]:BB:NFC:CBlock<ch>:FPGeneric:DLENGth?](#) on page 90

**Number Of Payload Bytes**

For "Technology > NFC A" sets the length of a standard frame.

For "Technology > NFC B /NFC F" sets the length of a frame.

Remote command:

[\[:SOURCE<hw>\]:BB:NFC:CBlock<ch>:APGeneric:STPLength](#) on page 91

[\[:SOURCE<hw>\]:BB:NFC:CBlock<ch>:BPGeneric:PLENGth](#) on page 91

[\[:SOURCE<hw>\]:BB:NFC:CBlock<ch>:FPGeneric:PLENGth](#) on page 91

**Number of Services**

Sets the number of services. The value determines the row numbers in the "Service Code List Configuration" dialog.

To access this dialog, select [Service Code List...](#)

Remote command:

[\[:SOURCE<hw>\]:BB:NFC:CBlock<ch>:NSERVICES](#) on page 104

**Number of Slots**

Determines number of slots (1, 2, 4, 8 or 16).

Remote command:

[\[:SOURCE<hw>\]:BB:NFC:CBlock<ch>:NOSLOTS](#) on page 104

**Number of Time Slots**

Determines how many time slots are used. The coding of the Time Slot Number TSN byte is performed accordingly.

Remote command:

[\[:SOURCE<hw>\]:BB:NFC:CBlock<ch>:TSN](#) on page 111

**A,B,E Parameter for MRTI (CHECK) or (UPDATE)**

Sets the value of the corresponding parameter, i.e. determines the format of the Maximum Response Time Information  $MRTI_{CHECK}$  and  $MRTI_{UPDATE}$ .

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:ACHK on page 87  
 [ :SOURce<hw> ] :BB:NFC:CBLock<ch>:AUPD on page 87  
 [ :SOURce<hw> ] :BB:NFC:CBLock<ch>:BCHK on page 87  
 [ :SOURce<hw> ] :BB:NFC:CBLock<ch>:BUPD on page 87  
 [ :SOURce<hw> ] :BB:NFC:CBLock<ch>:ECHK on page 87  
 [ :SOURce<hw> ] :BB:NFC:CBLock<ch>:EUPD on page 87

**Packet Selection**

Selects if the first or second packet of the SECTOR\_SELECT command is transmitted.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:PSElection on page 106

**PAD0, PAD1, PAD2**

Sets the value of PAD0/PAD1/PAD2 (hex).

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:PAD0 on page 105  
 [ :SOURce<hw> ] :BB:NFC:CBLock<ch>:PAD1 on page 105  
 [ :SOURce<hw> ] :BB:NFC:CBLock<ch>:PAD2 on page 105

**PDU Type**

Selects the type of PDU (Protocol Data Unit).

"Information"	Used to convey Application Layer Data in the transport data bytes. Application Layer Data is information for use by the adjacent upper layer.
"ACK/NACK"	Used to convey positive or negative acknowledgements. This PDU never contains transport data bytes. The acknowledgement relates to the last received PDU.
"Supervisory"	Used to exchange control information between the initiator and the target. Two different types of "Supervisory" PDUs are defined. For more details refer to the NFC Digital Protocol Technical Specification.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:PDUType on page 105

**PNI**

Only used with "PDU Type > Information". Determines Packet Number Information (PNI).

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:PNI on page 106

**Power Level Indication**

Determines the Power Level Indication.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:PLIN on page 105

### Power Level Indicator

Power Level Indicator.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:PLIR on page 105

### RC

Determines the Request Code (RC) ("No System Code info requested", "System Code info requested", "Advanced Protocol features supported") used to retrieve additional information.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:RC on page 106

### RTOX

With selected "PDU Type > Supervisory" and "ATN/Timeout > Timeout", sets the response timeout extension (RTOX) request value.

With a RTOX request, a target indicates that more time than the defined RWT is required to process the received PDU.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:RTOX on page 106

### SC

Sets the System Code.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:SCODE on page 107

### Segment Selection (ADD)

Selects a segment to be read.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:SEGSelection on page 107

### SEL\_CMD / SEL

Selects the cascade level (CL) of the NFCID1 / UID requested by the device.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:SCMD on page 107

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:SEL on page 107

### SEL\_PAR\_UPPER

Together with SEL\_PAR\_LOWER, the parameter SEL\_PAR\_UPPER determines where the NFC-A Bit oriented SDD Frame is split into the SDD\_REQ and SDD\_RES parts. Therefore this parameter influences the lengths of the SDD\_REQ or SDD\_RES commands. SEL\_PAR\_UPPER determines the number of full bytes of the SDD\_REQ part.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLock<ch>:SPUPper on page 109



**SEL\_PAR\_LOWER**

Together with SEL\_PAR\_UPPER, the parameter SEL\_PAR\_LOWER determines where the NFC-A Bit oriented SDD Frame is split into the SDD\_REQ and SDD\_RES parts. Therefore this parameter influences the lengths of the SDD\_REQ or SDD\_RES commands. SEL\_PAR\_LOWER determines the number of those bits of the SDD\_REQ part, which are not part of full bytes.

Remote command:

`[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:SPLower` on page 109

**Service Code List...**

Accesses the "Service Code List Configuration" dialog.

The number of rows corresponds to the value selected for the parameter [Number of Services](#).



"Access Attributes" Determines whether the attributes are "Read/Write" or "Read Only".

Remote command:

`[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:SERVICE<st>:AATTRIBUTES` on page 108

"Service Code" The "Service Code" is an element of the type 3 tags. Services are similar to files in a file system. Each service has a number of memory blocks associated with it. Services can be addressed using their service code, which must be unique inside each type 3 tag.

Remote command:

`[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:SERVICE<st>:SNUMBER` on page 108

**SFGI**

Determines the Start-up Frame Guard Time (SFGT).

Remote command:

`[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:SFGI` on page 109

**Slot Number**

Determines the slot number ("Slot Number 2" to "Slot Number 16"), i.e. defines the start of the response time slot during collision resolution.

Remote command:

`[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:SNUMBER` on page 109

**SNo**

For "Packet Selection > Packet 2", determines the sector number.

Remote command:

`[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:SNO` on page 109

**Status Flag 1, Status Flag 2**

Sets the status flags to specify a Type 3 tag's error condition. A value of 0 signals a successful execution, values different from 0 indicate errors.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:SF1 on page 108

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:SF2 on page 108

**Suppression of EoS, SoS Not Required**

Determines whether a suppression of EoS (End of Sequence)/SoS (Start of Sequence) is required or not.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:SENRequired on page 108

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:SSNRequired on page 110

**T1 to Tk**

For number of historical bytes *k* greater than 0, sets the historical bytes  $T_1$  to  $T_k$ .

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:T1TK on page 110

**Type 1 Tag Platform Configured**

Determines whether Type 1 Tag platform is configured or not.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:T1TConfigured on page 110

**Total No. Apps in the PICC**

Sets the total number of applications in the PICC (Proximity Inductive Coupling Card), i.e. in the NFC Forum Device in listener mode.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:TAIPicc on page 110

**WT**

Sets the Waiting Time (WT) that codes the Response Waiting Time (RWT). The "WT" value determines the least significant bits (b4 to b1) of the TO field in the ATR\_RES command.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:WT on page 111

**WTXM (INF field of S(WTX) request, response)**

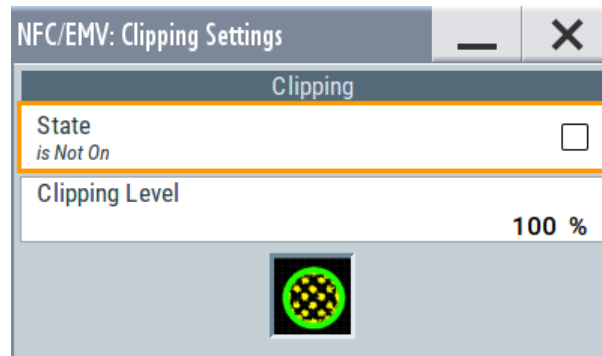
Only used when "DESELECT/WTX > WTX" is set. Sets the value of the WTXM in a waiting time extension request/response command.

Remote command:

[ :SOURce<hw> ] :BB:NFC:CBLOCK<ch>:WTXM on page 111

## 5.10 Clipping settings

- To access this dialog, select "General > Clipping Settings".



The dialog comprises the settings, necessary to configure the baseband filter, sample rate variation and clipping.

### 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.

With baseband clipping, the signal level is limited to a settable value ("Clipping Level"). This level is specified as a percentage of the highest peak value.

Remote command:

[\[:SOURce<hw>\]:BB:NFC:CLIPping:STATe](#) on page 115

### 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:NFC:CLIPping:LEVel](#) on page 115

## 6 How to generate signals with the NFC A/B/F option

The section provides examples on test setups for testing NFC enabled devices in polling and listening mode.

### 6.1 How to generate a signal for test of an NFC device in polling mode

A test setup for an NFC enabled mobile phone in polling mode, for testing carrier frequency, power level, modulation waveform and load modulation sensitivity requires a listener test signal ("Transmission Mode" = "Listen").

This is generated by the R&S SMW with option R&S SMW-K89.

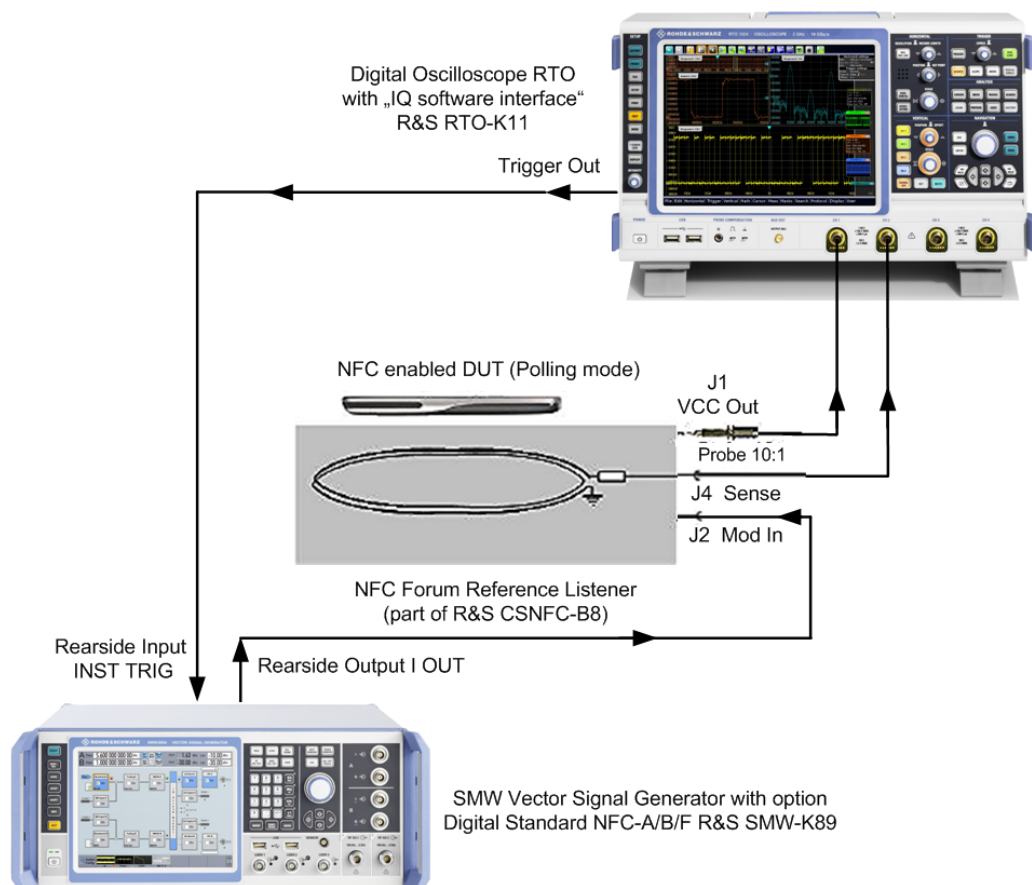


Figure 6-1: Test setup for an NFC mobile phone in polling mode (simplified schematic).

To generate the listener test signal proceed as follows:

1. Connect the I Out connector of the R&S SMW to the XJ2 (MOD IN) connector of the NFC Forum reference listener, see [Figure 6-1](#).
2. Provide an external trigger signal to the R&S SMW:  
Connect the trigger out connector of the measuring equipment (R&S RTO) to the input connector used for external triggering of digital modulations, standards and ARB of the R&S SMW.
3. Press the [Preset] hardkey at the R&S SMW.
4. Select the technology, for example NFC-B ("NFC/EMV > Technology > NFC-B").
5. Select the listen transmission mode ("NFC/EMV > Transmission Mode > Listen").
6. Select external triggering ("NFC/EMV > Trigger/Marker... > Trigger In > Mode > Single " and "NFC/EMV > Trigger/Marker... > Trigger In > Source > External Global Trigger" ).
7. Select the command type, for example "SENSB\_RES" ("NFC/EMV> Sequence Configuration > Command Type > SENSB\_RES"), see [Table 5-1](#) .
8. Activate the NFC signal ("NFC/EMV > State > On") .

If settings other than the default settings are required, add the following operating steps:

1. Set the modulation settings ("NFC/EMV > Modulation Settings"), see [Chapter 5.6, "Modulation settings"](#), on page 37.
2. Configure the frame ("NFC/EMV> Sequence Configuration > Frame Conf."), see [Chapter 5.9, "Frame configuration settings"](#), on page 50.
3. Set the clipping settings ("NFC/EMV > Clipping Settings"), see [Chapter 5.10, "Clipping settings"](#), on page 67.
4. Set the marker and clock settings ("NFC/EMV > Trigger/Marker..."), see [Chapter 5.2, "Trigger settings"](#), on page 28 and [Chapter 5.3, "Marker settings"](#), on page 33.
5. Set the parameter for "Desired Voltage in Unmodulated Signal parts" e.g. to 1.5 V ("NFC/EMV > Desired Voltage In Unmodulated Signal parts > 1.5 "), see [Chapter 3.3, "Leveling aspects"](#), on page 20.

## 6.2 How to generate a signal for test of an NFC device in listener mode

A test setup for an NFC enabled mobile phone in listener mode for test of load modulation, frame delay time etc., requires a poller test signal ("NFC A/B/F > Transmission Mode > Poll"). This is generated by the R&S SMW with option R&S SMW-K89.

## How to generate a signal for test of an NFC device in listener mode

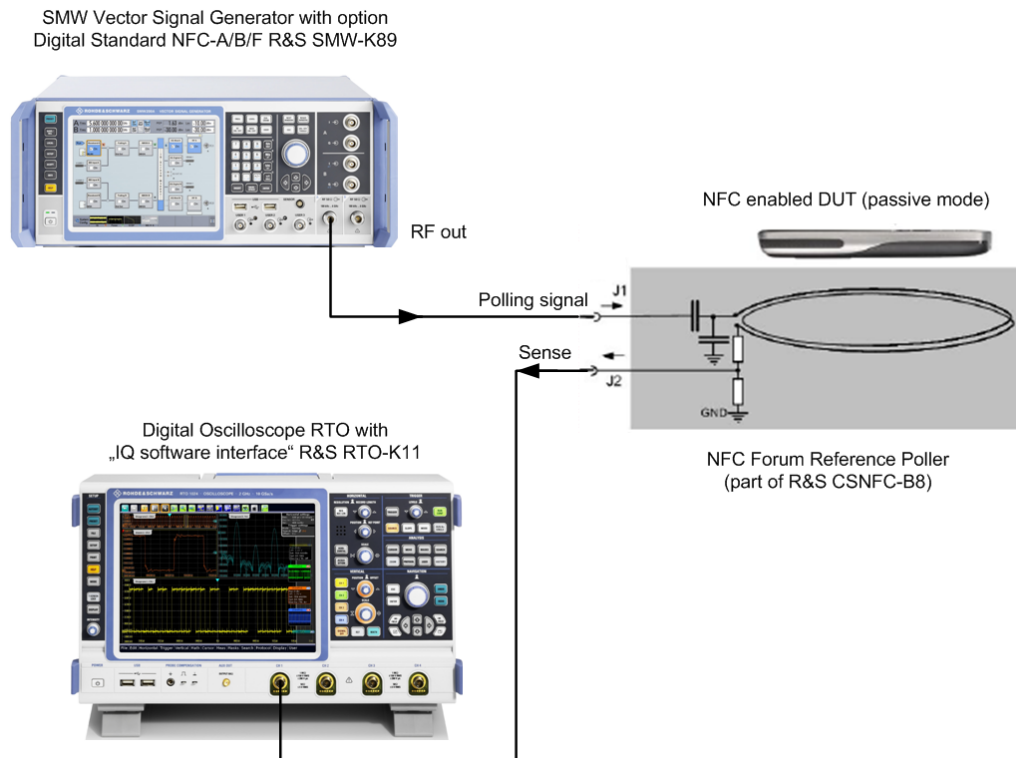


Figure 6-2: Test setup for an NFC mobile phone in listening mode (simplified schematic).

To generate the poller test signal proceed as follows.

1. Connect the RF A/RF B Out connector of the R&S SMW to the X1 connector of the NFC Forum reference poller, see [Figure 6-2](#).
2. Press the [Preset] hardkey at the R&S SMW.
3. Select the technology, for example NFC-B ("NFC/EMV > Technology > NFC-B").
4. Select the poll transmission mode ("NFC/EMV > Transmission Mode > Poll").
5. Select the command type, for example "ATTRIB" ("NFC/EMV > Sequence Configuration > Command Type > ATTRIB"), see [Table 5-1](#).
6. Activate the NFC signal ("NFC/EMV > State > On") .

If settings other than the default settings are required, add the following operating steps:

1. Set the modulation settings ("NFC/EMV > Modulation Settings"), see [Chapter 5.6, "Modulation settings"](#), on page 37.
2. Configure the frame ("NFC/EMV > Sequence Configuration > Frame Conf."), see [Chapter 5.9, "Frame configuration settings"](#), on page 50.
3. Set the clipping settings ("NFC/EMV > Clipping Settings"), see [Chapter 5.10, "Clipping settings"](#), on page 67.

4. Set the marker and clock settings ("NFC/EMV > Trigger/Marker..."), see [Chapter 5.2, "Trigger settings"](#), on page 28 and [Chapter 5.3, "Marker settings"](#), on page 33.

## 7 Remote-control commands

The following commands are required to perform signal generation with the NFC option in a remote environment. We assume that the R&S SMW has already been set up for remote operation in a network as described in the R&S SMW documentation. A knowledge about the remote control operation and the SCPI command syntax are 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 SMW user manual.

### Common Suffixes

The following common suffixes are used in remote commands:

Suffix	Value range	Description
ENTity<ch>	1 to 4	entity in a multiple entity configuration with separate baseband sources ENTity3 4 require option R&S SMW-K76
SOURce<hw>	[1] to 4	available baseband signals only SOURce1 possible, if the keyword ENTity is used
OUTPut<ch>	1 to 3	available markers
CBLOCK<ch>	1 .. 100	successive number of the command block in the sequence configuration table
BLOCK<st>	1 .. 100	Index of the entry in the "Block List Table"
SERVice<st>	1 .. 100	Index of the entry in the "Service List Table"



### Using SCPI command aliases for advanced mode with multiple entities

You can address multiple entities configurations by using the SCPI commands starting with the keyword SOURce or the alias commands starting with the keyword ENTity.

Note that the meaning of the keyword SOURce<hw> changes in the second case.

For details, see section "SCPI Command Aliases for Advanced Mode with Multiple Entities" in the R&S SMW user manual.

The following commands specific to the NFC are described here:

- [Programming example](#).....73
- [Primary settings](#).....76
- [Save/recall operations](#).....78
- [Predefined sequence](#).....79
- [Sequence configuration](#).....81
- [Frame configuration](#).....84
- [Modulation settings](#).....112
- [Clipping settings](#).....115



• <a href="#">Trigger settings</a> .....	116
• <a href="#">Marker settings</a> .....	121
• <a href="#">Clock settings</a> .....	123

## 7.1 Programming example

With the following programm the configuration as described in [Chapter 6, "How to generate signals with the NFC A/B/F option"](#), on page 68 is set at the instrument via remote control. Note that different setups and remote command sequences are required for listener and poller test signal.

The settings 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 most remote control programs, an instrument (p)reset is recommended to set the R&S SMW to a definite state. The commands `*RST` and `SYSTem:PRESet` are equivalent for this purpose. `*CLS` resets the status registers and clears the output buffer.

In the example we assume that a remote PC is connected to the instrument, the remote PC and the instrument are switched on and a connection between them is established.

The other connections are setup as shown in [Chapter 6, "How to generate signals with the NFC A/B/F option"](#), on page 68

### Example: Generate a listener test signal

The following example generates a listener test signal.

```
// *****
// Reset NFC settings
// Query the used NFC FW-Version.
// Response:"NFCForum-TS-DigitalProtocol-1.0 NFCForum-TS-Analog-1.0
// Set technology to NFC-B
// Set transmission mode to listen
// For signal generators only: Select external trigger source at the
// connector
// *****
*RST
SOURce:BB:NFC:PRESet
*CLS
SOURce:BB:NFC:VERSion?
SOURce:BB:NFC:TECHnology NFCB
SOURce:BB:NFC:TMode LISTEN
SOURce:BB:NFC:TRIGger:SOURce EGT1

// *****
// Select command type SENSB_RES and activate NFC signal
```

```
// *****
SOURCE:BB:NFC:CBlock1:CTYPE SNBS
SOURCE:BB:NFC:STATE 1
```

If settings other than the default settings are required, add the following operating steps:

```
// *****
// Set modulation settings
// Activate slope
// Set risetime to 0.25 us, other modulation settings stay default
// For signal generators only: Set the "desired voltage in unmodulated
// signal parts"
// *****
SOURCE:BB:NFC:MSET:SLOPe 1
SOURCE:BB:NFC:MSET:TRISe .25

// *****
// Configure the frame (for command type SENSB_RES)
// Set application data to "2345 ABCD" (Hex)
// *****
SOURCE:BB:NFC:CBlock1:ADATa #H2345ABCD, 32

// *****
// For signal generators only:
// Set the "Desired Voltage in Unmodulated Signal parts" to 1.5 Volts.
// Cause the instrument to automatically adjust the related parameters
// of the analog I and Q outputs
// *****
SOURCE:BB:NFC:DVOLTage 1.5
SOURCE:BB:NFC:UAISetting
```

**Example: Generate a poller test signal**

The following example generates a poller test signal.

```
// *****
// Set technology to NFC-B
// Set transmission mode to poll
// Select command type ATTRIB and activate NFC signal
// *****
*RST
*CLS
SOURCE:BB:NFC:TECHnology NFCB
SOURCE:BB:NFC:TMODe POLL
SOURCE:BB:NFC:CBLock1:CTYPe ATBQ
SOURCE:BB:NFC:STATe ON

// Alternatively set the predefined NFC-A sequence APA
// (IDLE, ALL_REQ, IDLE, BLANK)
// For signal generators only: position marker1 after first idle
// Activate the sequence and marker (if applicable)
*RST
*CLS
SOURCE:BB:NFC:TMODe POLL
SOURCE:BB:NFC:PRED:SEquence APA
SOURCE:BB:NFC:PRED:CNFMarker ON
SOURCE:BB:NFC:PRED:APPLy

// *****
// Set technology to NFC-B
// Set transmission mode to poll
// Select command type ATTRIB and activate NFC signal
// *****
*RST
*CLS
SOURCE:BB:NFC:TECHnology NFCB
SOURCE:BB:NFC:TMODe POLL
SOURCE:BB:NFC:CBLock1:CTYPe ATBQ
SOURCE:BB:NFC:STATe ON

// Alternatively set the predefined NFC-A sequence APA
// (IDLE, ALL_REQ, IDLE, BLANK)
// For signal generators only: position marker1 after first idle
// Activate the sequence and marker (if applicable)
*RST
*CLS
SOURCE:BB:NFC:TMODe POLL
SOURCE:BB:NFC:PRED:SEquence APA
SOURCE:BB:NFC:PRED:CNFMarker ON
SOURCE:BB:NFC:PRED:APPLy
```

If settings other than the default settings are required, add the following operating steps to the set predefined NFC-A sequence:

```
// *****
// Configure the sequence and the frame:
// Set command type of first sequence-command to "DATA_Type4A"
// Set the block type to I-Block
// Set the data length to 2 bytes
// Set the data pattern 0110 0101 0011 1100
// *****
SOURce:BB:NFC:CBlock1:CTYPE T4AD
SOURce:BB:NFC:CBlock1:BTYPe TPI
SOURce:BB:NFC:CBlock1:DATA:LENGth 2
SOURce:BB:NFC:CBlock1:DATA:PATtern #B0110010100111100, 16
```

## 7.2 Primary settings

<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:DIVisor.....</a>	76
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:DVOLtage.....</a>	76
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:PRESet.....</a>	77
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:STATe.....</a>	77
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TECHnology.....</a>	77
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TMODe.....</a>	77
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:UAISetting.....</a>	77
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:UPVoltage?.....</a>	78
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:VERSion?.....</a>	78
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:WAVEform:CREate.....</a>	78

---

### **[:SOURce<hw>]:BB:NFC:DIVisor <DivForMod>**

Selects the divisor and thus the datarate for technology NFC-F.

#### **Parameters:**

<DivForMod>            DIV2 | DIV4  
                          \*RST:        DIV2

**Manual operation:**    See ["Divisor \(Bit Rate\)"](#) on page 26

---

### **[:SOURce<hw>]:BB:NFC:DVOLtage <DVoltage>**

Sets the desired voltage in unmodulated signal parts.

#### **Parameters:**

<DVoltage>            float  
                          Range:        0.020 to 1.5  
                          Increment: 0.001  
                          \*RST:        1

**Example:**             See [Chapter 7.1, "Programming example"](#), on page 73

**Manual operation:**    See ["Desired Voltage In Unmodulated Signal parts"](#) on page 27

---

**[[:SOURce<hw>]:BB:NFC: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<hw>:BB:NFC:STATe`.

**Example:** See [Chapter 7.1, "Programming example"](#), on page 73

**Usage:** Event

**Manual operation:** See ["Set to Default"](#) on page 25

---

**[[:SOURce<hw>]:BB:NFC: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 [Chapter 7.1, "Programming example"](#), on page 73

**Manual operation:** See ["State"](#) on page 24

---

**[[:SOURce<hw>]:BB:NFC:TECHnology <Protocol>**

Selects the NFC/EMV technology.

**Parameters:**

<Protocol> NFCA | NFCB | NFCF | EMVA | EMVB  
\*RST: NFCA

**Example:** See [Chapter 7.1, "Programming example"](#), on page 73.

**Manual operation:** See ["Technology"](#) on page 26

---

**[[:SOURce<hw>]:BB:NFC:TMODe <Transmission>**

Selects the transmission mode.

**Parameters:**

<Transmission> POLL | LISTen  
\*RST: POLL

**Example:** See [Chapter 7.1, "Programming example"](#), on page 73.

**Manual operation:** See ["Transmission Mode"](#) on page 26

---

**[[:SOURce<hw>]:BB:NFC:UAISetting**

Triggers the instrument to automatically adjust the related parameters of the analog I and Q outputs.

**Example:** See [Chapter 7.1, "Programming example"](#), on page 73

**Manual operation:** See ["Update Analog I/Q Settings For Desired Voltage"](#) on page 27

#### **[ :SOURce<hw>]:BB:NFC:UPVoltage?**

Displays the ratio of the voltage in the unmodulated parts of the signal to its peak value.

##### **Return values:**

<UPVoltage>            integer  
                              Range:        0 to 100  
                              \*RST:        0

**Usage:**                Query only

**Manual operation:** See ["Unmodulated Parts Voltage To Peak Voltage Ratio"](#) on page 27

#### **[ :SOURce<hw>]:BB:NFC:VERSion?**

Queries the version of the NFC-Forum and EMVCo specifications used for the signal generation.

##### **Return values:**

<Version>               string

**Example:**               See [Chapter 7.1, "Programming example"](#), on page 73.

**Usage:**                Query only

#### **[ :SOURce<hw>]:BB:NFC:WAVEform:CREate <Filename>**

Stores the current NFC signal as ARB signal in a waveform file with the filename given in the parameter.

##### **Setting parameters:**

<Filename>               string

**Usage:**                Setting only

**Manual operation:** See ["Generate Waveform File..."](#) on page 26

## 7.3 Save/recall operations

<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:SETTing:CATalog?</a> .....	79
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:SETTing:DELeTe</a> .....	79
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:SETTing:LOAD</a> .....	79
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:SETTing:STORe</a> .....	79

**[[:SOURce<hw>]:BB:NFC:SETTing:CATalog?**

Catalog settings file name.

**Return values:**

<Catalog>                      string

**Usage:**                              Query only

**Manual operation:**    See ["Save/Recall ..."](#) on page 25

**[[:SOURce<hw>]:BB:NFC:SETTing:DELeTe <Filename>**

Deletes the NFC settings file with the filename given in the parameter.

**Setting parameters:**

<Filename>                      string

**Usage:**                              Setting only

**Manual operation:**    See ["Save/Recall ..."](#) on page 25

**[[:SOURce<hw>]:BB:NFC:SETTing:LOAD <Filename>**

Loads the NFC setting file with the name given in the parameter.

**Setting parameters:**

<Filename>                      string

**Usage:**                              Setting only

**Manual operation:**    See ["Save/Recall ..."](#) on page 25

**[[:SOURce<hw>]:BB:NFC:SETTing:STORe <Filename>**

Stores current NFC settings in a file with the name given in the parameter.

**Setting parameters:**

<Filename>                      string

**Usage:**                              Setting only

**Manual operation:**    See ["Save/Recall ..."](#) on page 25

## 7.4 Predefined sequence

<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:PRED:SEQuence.....</a>	80
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:PRED:CNFMarker.....</a>	80
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:PRED:APPLy.....</a>	80

---

**[[:SOURce<hw>]:BB:NFC:PRED:SEQUence <Sequence>**

Available only for "Transmission Mode > Poll" and "Transmission Mode > PCD to PICC".

Selects a predefined sequence.

**Parameters:**

<Sequence>

FPS | BPA | BPS | APA | APS

**FPS**

Predefined NFC-F sequence with the elements: IDLE, SENSF\_REQ, IDLE, BLANK

**BPA**

Predefined NFC-B sequence with the elements: IDLE, ALL\_REQ, IDLE, BLANK or a predefined EMV Type A sequence with the elements: IDLE, WUPB, IDLE, BLANK

**BPS**

Predefined NFC-B sequence with the elements: IDLE, SENS\_REQ, IDLE, BLANK or a predefined EMV Type B sequence with the elements: IDLE, REQB, IDLE, BLANK

**APA**

Predefined NFC-A sequence with the elements: IDLE, ALL\_REQ, IDLE, BLANK or a predefined EMV Type A sequence with the elements: IDLE, WUPA, IDLE, BLANK

**APS**

Predefined NFC-A sequence with the elements: IDLE, SENS\_REQ, IDLE, BLANK or a predefined EMV Type A sequence with the elements: IDLE, REQA, IDLE, BLANK

\*RST: APS

**Example:** See [Chapter 7.1, "Programming example"](#), on page 73.

**Manual operation:** See ["Predefined Sequence"](#) on page 43

---

**[[:SOURce<hw>]:BB:NFC:PRED:CNFMarker <Conf>**

Available for signal generators only.

If enabled marker 1 is positioned after the first idle.

**Parameters:**

<Conf>

1 | ON | 0 | OFF

**Example:** See [Chapter 7.1, "Programming example"](#), on page 73

.

**Manual operation:** See ["Configure Marker 1 For The Sequence"](#) on page 43

---

**[[:SOURce<hw>]:BB:NFC:PRED:APPLY**

Activates the selected "Predefined Sequence" and marker.



**Example:** See [Chapter 7.1, "Programming example"](#), on page 73  
.

**Usage:** Event

**Manual operation:** See ["Apply"](#) on page 43

## 7.5 Sequence configuration

<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLOCK:APPend</a> .....	81
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:COPY</a> .....	81
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CCBLOCK</a> .....	81
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:CTYPE</a> .....	82
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:DELeTe</a> .....	82
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:DCBLOCK</a> .....	82
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:DURation</a> .....	82
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:INSert</a> .....	82
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:ICBLOCK</a> .....	82
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:PASTe</a> .....	83
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:PCBLOCK</a> .....	83
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:POFFset</a> .....	83
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:REPetition</a> .....	83
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:SAMPles?</a> .....	83
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:STIME?</a> .....	84
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:SCONfiguration:TNSamples?</a> .....	84
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:SCONfiguration:TSDuration?</a> .....	84

---

### **[\[:SOURce<hw>\]:BB:NFC:CBLOCK:APPend](#)**

Appends a command block to the end of the command sequence.

**Usage:** Event

**Manual operation:** See ["Append, Insert, Delete, Copy, Paste"](#) on page 49

---

### **[\[:SOURce<hw>\]:BB:NFC:CBLOCK<ch>:COPY](#)**

**[\[:SOURce<hw>\]:BB:NFC:CCBLOCK](#) <CcBlock>**

Copies a command block for later use.

#### **Setting parameters:**

<CcBlock> integer  
Range: 1 to 100  
\*RST: 1

**Usage:** Setting only

**Manual operation:** See ["Append, Insert, Delete, Copy, Paste"](#) on page 49

---

**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:CTYPE <Cmd>**

Selects the command type.

**Parameters:**

<Cmd>                    ALAQ | SNAQ | SDAQ | SLAQ | SPAQ | RDAQ | RLAQ | T1RQ |  
                           WREQ | WNEQ | RSGQ | RD8Q | WE8Q | WN8Q | T2RQ |  
                           T2WQ | SSLQ | RATQ | T4AD | ATRQ | PSLQ | DEPQ | DSLQ |  
                           RLSQ | ALBQ | SNBQ | SMAR | SPBQ | ATBQ | T4BD | SNFQ |  
                           CHKQ | UPDQ | SNAS | SDAS | SLAS | RDAS | RLAS | T1RS |  
                           WRES | WNES | RSGS | RD8S | WE8S | WN8S | T2RS | ACK |  
                           NACK | ATSS | ATRS | PSLS | DEPS | DSLS | RLSS | SNBS |  
                           SPBS | ATBS | SNFS | CHKS | UPDS | GENE | IDLE | BLNK  
                           \*RST:        SNAQ

**Example:**                See [Chapter 7.1, "Programming example"](#), on page 73.

**Manual operation:**    See ["Command Type"](#) on page 45

For command types and the corresponding SCPI-command parameters see the overview in [Table 5-1](#).

---

**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:DELETE**

**[[:SOURce<hw>]:BB:NFC:DCBLOCK <DcBlock>**

Removes a command block from the command sequence.

**Setting parameters:**

<DcBlock>                integer  
                              Range:        1 to 100

**Usage:**                    Setting only

**Manual operation:**    See ["Append, Insert, Delete, Copy, Paste"](#) on page 49

---

**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:DURATION <Duration>**

Determines the frame period in  $\mu$ s.

**Parameters:**

<Duration>                float  
                              Range:        0 to 1E6  
                              Increment: 0.01  
                              \*RST:        84.9557522

**Manual operation:**    See ["Duration \( \$\mu\$ s\)"](#) on page 49

---

**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:INSERT**

**[[:SOURce<hw>]:BB:NFC:ICBLOCK <IcBlock>**

Inserts a default command block before the selected command block. The command block with this position must be existing, otherwise an error is returned.

**Setting parameters:**

<IcBlock> integer  
Range: 1 to 99

**Usage:** Setting only

**Manual operation:** See ["Append, Insert, Delete, Copy, Paste"](#) on page 49

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:PASTe**

**[ :SOURce<hw>]:BB:NFC:PCBLock <PcBlock>**

Pastes a command block (which was copied before) at the given position into the command sequence.

**Setting parameters:**

<PcBlock> integer  
Range: 1 to 99

**Usage:** Setting only

**Manual operation:** See ["Append, Insert, Delete, Copy, Paste"](#) on page 49

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:POFFset <OFFSet>**

Determines the power offset value in dB.

**Parameters:**

<OFFSet> float  
Range: -20 to 20  
Increment: 0.01  
\*RST: 0

**Manual operation:** See ["Power Offset \(dB\)"](#) on page 49

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:REPetition <Repet>**

Determines the number of times to repeat the generation of a frame.

**Parameters:**

<Repet> integer  
Range: 0 to 9999  
\*RST: 1

**Manual operation:** See ["Rep."](#) on page 49

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:SAMPles?**

Queries the total number of samples in a selected command block.

**Return values:**

<Samples> float

**Usage:** Query only

**Manual operation:** See ["Samples"](#) on page 49

---

#### **[[:SOURce<hw>]:BB:NFC:CBLock<ch>:STIME?**

Queries the exact start time of the corresponding command.

**Return values:**

<STIME> float

**Usage:** Query only

**Manual operation:** See ["Start Time"](#) on page 45

---

#### **[[:SOURce<hw>]:BB:NFC:SCONfiguration:TNSamples?**

Queries the total number of samples allocated to the current frame.

**Return values:**

<TNSamples> integer

**Usage:** Query only

**Manual operation:** See ["Total Number of Samples"](#) on page 44

---

#### **[[:SOURce<hw>]:BB:NFC:SCONfiguration:TSDuration?**

Queries the total sequence duration for the current settings.

**Return values:**

<TSDuration> float

**Usage:** Query only

**Manual operation:** See ["Total Sequence Duration"](#) on page 44

## 7.6 Frame configuration

<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLock&lt;ch&gt;:ACHK</a>	87
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLock&lt;ch&gt;:AUPD</a>	87
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLock&lt;ch&gt;:BCHK</a>	87
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLock&lt;ch&gt;:BUPD</a>	87
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLock&lt;ch&gt;:ECHK</a>	87
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLock&lt;ch&gt;:EUPD</a>	87
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLock&lt;ch&gt;:ADATa</a>	87
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLock&lt;ch&gt;:ADCoding</a>	87
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLock&lt;ch&gt;:AFI</a>	88
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLock&lt;ch&gt;:AID</a>	88
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLock&lt;ch&gt;:ALENgtH</a>	88
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLock&lt;ch&gt;:ANSelection</a>	88
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLock&lt;ch&gt;:APFSupported</a>	88
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:CBLock&lt;ch&gt;:APGeneric:BOData</a>	89

[SOURce<hw>]:BB:NFC:CBLOCK<ch>:APGeneric:BOLength.....	89
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:APGeneric:FTYPE.....	89
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:APGeneric:SHData.....	89
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:APGeneric:SHLength.....	90
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BPGeneric:DATA.....	90
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:FPGeneric:DATA.....	90
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:APGeneric:STData.....	90
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BPGeneric:DLENGth?.....	90
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:FPGeneric:DLENGth?.....	90
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:APGeneric:STDLength?.....	90
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BPGeneric:PLENGth.....	91
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:FPGeneric:PLENGth.....	91
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:APGeneric:STPLength.....	91
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:APGeneric:STEPresent.....	91
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:ATIMEout.....	91
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BCCError.....	91
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BFANticollision.....	91
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BFSDd.....	92
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BLOCK<st>:BDATa.....	92
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BLKSelection.....	92
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BLOCK<st>:BNUmber.....	92
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BLOCK<st>:LEN.....	92
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BLOCK<st>:LOCKed.....	93
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BLOCK<st>:SLORder.....	93
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BNO.....	93
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BTYPE.....	93
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BYTSelection.....	94
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:CFGType.....	94
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:CHAIning.....	94
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:CID.....	94
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:CIIL4.....	94
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:CSUPported.....	95
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DATA.....	95
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DATA:DSElection.....	95
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DATA:LENGth.....	95
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DATA:PATtern.....	95
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DEQD.....	96
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DFOLlowing.....	96
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DID.....	96
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DLP2.....	96
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DLP4.....	97
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DLP8.....	97
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DLTPoll.....	97
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DPL2.....	97
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DPL4.....	97
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DPL8.....	98
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DPPCd.....	98
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DPPicc.....	98
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DPTListen.....	98
[SOURce<hw>]:BB:NFC:CBLOCK<ch>:DRI.....	98

[SOURce<hw>]:BB:NFC:CBLock<ch>:DSI.....	99
[SOURce<hw>]:BB:NFC:CBLock<ch>:DSUPported.....	99
[SOURce<hw>]:BB:NFC:CBLock<ch>:DWSelection.....	99
[SOURce<hw>]:BB:NFC:CBLock<ch>:EASupported.....	99
[SOURce<hw>]:BB:NFC:CBLock<ch>:ESSupported.....	99
[SOURce<hw>]:BB:NFC:CBLock<ch>:FSC.....	99
[SOURce<hw>]:BB:NFC:CBLock<ch>:FWI.....	100
[SOURce<hw>]:BB:NFC:CBLock<ch>:GBSelection.....	100
[SOURce<hw>]:BB:NFC:CBLock<ch>:GDAAvailable.....	100
[SOURce<hw>]:BB:NFC:CBLock<ch>:IBNumber.....	100
[SOURce<hw>]:BB:NFC:CBLock<ch>:KPARameter.....	101
[SOURce<hw>]:BB:NFC:CBLock<ch>:LREDuction.....	101
[SOURce<hw>]:BB:NFC:CBLock<ch>:MBLI.....	101
[SOURce<hw>]:BB:NFC:CBLock<ch>:MFSize.....	101
[SOURce<hw>]:BB:NFC:CBLock<ch>:MICHaining.....	101
[SOURce<hw>]:BB:NFC:CBLock<ch>:MTR0.....	102
[SOURce<hw>]:BB:NFC:CBLock<ch>:MTR1.....	102
[SOURce<hw>]:BB:NFC:CBLock<ch>:MTR2.....	102
[SOURce<hw>]:BB:NFC:CBLock<ch>:N2FTType.....	102
[SOURce<hw>]:BB:NFC:CBLock<ch>:NACK.....	102
[SOURce<hw>]:BB:NFC:CBLock<ch>:NAD.....	103
[SOURce<hw>]:BB:NFC:CBLock<ch>:NBLocks.....	103
[SOURce<hw>]:BB:NFC:CBLock<ch>:NFOLlowing.....	103
[SOURce<hw>]:BB:NFC:CBLock<ch>:NID0.....	103
[SOURce<hw>]:BB:NFC:CBLock<ch>:NID1.....	103
[SOURce<hw>]:BB:NFC:CBLock<ch>:NID2.....	103
[SOURce<hw>]:BB:NFC:CBLock<ch>:NNComplete.....	104
[SOURce<hw>]:BB:NFC:CBLock<ch>:NOAPplications.....	104
[SOURce<hw>]:BB:NFC:CBLock<ch>:NOSLots.....	104
[SOURce<hw>]:BB:NFC:CBLock<ch>:NSERVICES.....	104
[SOURce<hw>]:BB:NFC:CBLock<ch>:NSIZE.....	104
[SOURce<hw>]:BB:NFC:CBLock<ch>:NSUPported.....	105
[SOURce<hw>]:BB:NFC:CBLock<ch>:PAD0.....	105
[SOURce<hw>]:BB:NFC:CBLock<ch>:PAD1.....	105
[SOURce<hw>]:BB:NFC:CBLock<ch>:PAD2.....	105
[SOURce<hw>]:BB:NFC:CBLock<ch>:PDUType.....	105
[SOURce<hw>]:BB:NFC:CBLock<ch>:PLIN.....	105
[SOURce<hw>]:BB:NFC:CBLock<ch>:PLIR.....	105
[SOURce<hw>]:BB:NFC:CBLock<ch>:PNI.....	106
[SOURce<hw>]:BB:NFC:CBLock<ch>:PSElection.....	106
[SOURce<hw>]:BB:NFC:CBLock<ch>:PUPI.....	106
[SOURce<hw>]:BB:NFC:CBLock<ch>:RC.....	106
[SOURce<hw>]:BB:NFC:CBLock<ch>:RTOX.....	106
[SOURce<hw>]:BB:NFC:CBLock<ch>:SCMD.....	107
[SOURce<hw>]:BB:NFC:CBLock<ch>:SCODE.....	107
[SOURce<hw>]:BB:NFC:CBLock<ch>:SEGSelection.....	107
[SOURce<hw>]:BB:NFC:CBLock<ch>:SEL.....	107
[SOURce<hw>]:BB:NFC:CBLock<ch>:SENRequired.....	108
[SOURce<hw>]:BB:NFC:CBLock<ch>:SERvice<st>:AATTributes.....	108
[SOURce<hw>]:BB:NFC:CBLock<ch>:SERvice<st>:SNUMber.....	108

<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:SF1</code>	108
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:SF2</code>	108
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:SFGI</code>	109
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:SNO</code>	109
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:SNUMber</code>	109
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:SPLower</code>	109
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:SPUPper</code>	109
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:SSNRequired</code>	110
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:T1TConfigured</code>	110
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:T1TK</code>	110
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:TAIPicc</code>	110
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:TSN</code>	111
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:UID</code>	111
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:UNComplete</code>	111
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:USIZe</code>	111
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:WT</code>	111
<code>[SOURce&lt;hw&gt;]:BB:NFC:CBLOCK&lt;ch&gt;:WTXM</code>	111

---

`[SOURce<hw>]:BB:NFC:CBLOCK<ch>:ACHK <ACheck>`  
`[SOURce<hw>]:BB:NFC:CBLOCK<ch>:AUPD <AUpdate>`  
`[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BCHK <BCheck>`  
`[SOURce<hw>]:BB:NFC:CBLOCK<ch>:BUPD <BUpdate>`  
`[SOURce<hw>]:BB:NFC:CBLOCK<ch>:ECHK <ECheck>`  
`[SOURce<hw>]:BB:NFC:CBLOCK<ch>:EUPD <EUpdate>`

Determines the format and value of the Maximum Response Time Information  
MRTI<sub>CHECK</sub> and MRTI<sub>UPDATE</sub>.

**Parameters:**

`<EUpdate>` integer  
Range: 0 to 3  
\*RST: 0

**Manual operation:** See "A,B,E Parameter for MRTI (CHECK) or (UPDATE)" on page 63

---

`[SOURce<hw>]:BB:NFC:CBLOCK<ch>:ADATa <AData>`

Application data input (hex value).

**Parameters:**

`<AData>` integer

**Example:** See Chapter 7.1, "Programming example", on page 73

**Manual operation:** See "Application Data" on page 52

---

`[SOURce<hw>]:BB:NFC:CBLOCK<ch>:ADCoding <ADCoding>`

Determines if application is proprietary or CRC-B.

**Parameters:**

<ADCoding>            PROP | CRCB  
                          \*RST:        PROP

**Manual operation:**    See ["Application Data Coding"](#) on page 52

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:AFI <Afi>**

Sets the application family being selected.

**Parameters:**

<Afi>                    integer  
                          Range:        0 to 255  
                          \*RST:        0

**Manual operation:**    See ["AFI"](#) on page 52

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:AID <Aid>**

Determines the value of AID.

**Parameters:**

<Aid>                    integer

**Manual operation:**    See ["AID \(hex\)"](#) on page 52

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:ALENgtH <ALength>**

Determines the length of AID.

**Parameters:**

<ALength>              integer  
                          Range:        1 to 16  
                          \*RST:        1

**Manual operation:**    See ["AID Length"](#) on page 52

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:ANSelection <ANSelection>**

Available only for "PDU Type > ACK-NACK" or "Block Type > R-block".

Selects ACK or NACK.

**Parameters:**

<ANSelection>           ACK | NACK  
                          \*RST:        ACK

**Manual operation:**    See ["ACK, NACK"](#) on page 51

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:APFSupported <APFS>**

Determines if Advanced Protocol Features are supported.



**Parameters:**

<APFS> 1 | ON | 0 | OFF  
 \*RST: 0

**Manual operation:** See ["Advanced Protocol Features supported"](#) on page 51

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:BOData <BoFrameData>**

Sets the data for a bit oriented SDD frame.

**Parameters:**

<BoFrameData> integer  
 \*RST: #H0000

**Example:**

```
:BB:NFC:CBL1:APG:BOL 18
sets the length of the standard frame to 18 bits
:BB:NFC:CBL1:APG:BOD #H3FFFF,18
sets the data
```

**Manual operation:** See ["Data \(hex\)"](#) on page 55

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:BOLength <BoFrameLen>**

Sets the length of the first part of a bit oriented SDD frame.

**Parameters:**

<BoFrameLen> integer  
 Range: 16 to 55  
 \*RST: 16

**Manual operation:** See ["Number Of Bits For Part 1"](#) on page 61

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:FTYPE <FrameType>**

Selects a frame type for "Command Type > GENERIC".

**Parameters:**

<FrameType> SHORT | STANdard | BOSDd  
 \*RST: SHORT

**Manual operation:** See ["Frame Type"](#) on page 58

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:SHData <ShortFrameData>**

Sets the data bits of a short frame.

**Parameters:**

<ShortFrameData> integer  
 \*RST: #H00

**Example:**                   :BB:NFC:CBL1:APG:SHL 7  
                               sets the length of the short frame to 7 bits  
                               :BB:NFC:CBL1:APG:SHD #H26,7  
                               sets the data

**Manual operation:**   See ["Data \(hex\)"](#) on page 55

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:SHLength <ShortFrameLen>**

Sets the length of a short frame in bits.

**Parameters:**

<ShortFrameLen>   integer  
                           Range:       1 to 7  
                           \*RST:       7

**Manual operation:**   See ["Number Of Bits"](#) on page 61

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:BPGeneric:DATA <Data>**  
**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:FPGeneric:DATA <Data>**  
**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:STDData <StdFrameData>**

Sets the data for a standard frame in hexadecimal values.

**Parameters:**

<StdFrameData>   integer

**Example:**                   :BB:NFC:CBL1:APG:STPL 4  
                               sets the length of the standard frame to 4 bytes  
                               :BB:NFC:CBL1:APG:STD #H01234567,32  
                               sets the data

**Manual operation:**   See ["Data \(hex\)"](#) on page 55

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:BPGeneric:DLEngh? <DataLength>**  
**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:FPGeneric:DLEngh? <DataLength>**  
**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:STDLength? <StdFrameDataLen>**

Shows the total length of a standard frame in bytes.

**Parameters:**

<StdFrameDataLen> integer  
                           Range:       1 to 10  
                           \*RST:       3

**Usage:**                   Query only

**Manual operation:**   See ["Number Of Data Bytes"](#) on page 62

---

```
[ :SOURce<hw>]:BB:NFC:CBLock<ch>:BPGeneric:PLENgtH <PayloadLength>
[ :SOURce<hw>]:BB:NFC:CBLock<ch>:FPGeneric:PLENgtH <PayloadLength>
[ :SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:STPLenGth <StdFramePayLen>
```

Sets the length of a standard frame.

**Parameters:**

```
<StdFramePayLen> integer
                    Range:    1 to 8
                    *RST:     1
```

**Manual operation:** See ["Number Of Payload Bytes"](#) on page 62

---

```
[ :SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:STEPresent
                    <StdFrameEodPres>
```

Selects if the EoD is present or not.

**Parameters:**

```
<StdFrameEodPres> 1 | ON | 0 | OFF
                    *RST:     1
```

**Manual operation:** See ["EoD \(CRC\)"](#) on page 58

---

```
[ :SOURce<hw>]:BB:NFC:CBLock<ch>:ATIMEout <ATimeout>
```

Only used with PDU type "supervisory". Determines whether an "ATN" (Attention) or "Timeout" supervisory PDU type is used.

**Parameters:**

```
<ATimeout>         ATN | TOUT
                    *RST:     ATN
```

**Manual operation:** See ["ATN or Timeout"](#) on page 52

---

```
[ :SOURce<hw>]:BB:NFC:CBLock<ch>:BCCError <BCCError>
```

If enabled, an error is added intentionally to the BCC (checksum).

**Parameters:**

```
<BCCError>         1 | ON | 0 | OFF
                    *RST:     0
```

**Manual operation:** See ["BCC Error"](#) on page 52

---

```
[ :SOURce<hw>]:BB:NFC:CBLock<ch>:BFANTicollision <BFANTicol>
```

Determines the bit frame Anticollision.

**Parameters:**

```
<BFANTicol>        SDD0 | SDD2 | SDD1 | SDD4 | SDD8 | SDD16
```

**Manual operation:** See ["Bit Frame SDD / Bit Frame Anticollision"](#) on page 52

---

**[ :SOURce<hw>]:BB:NFC:CBLOCK<ch>:BFSDd <BFSdd>**

Determines Bit frame SDD.

**Parameters:**

<BFSdd> SDD0 | SDD2 | SDD1 | SDD4 | SDD8 | SDD16  
 \*RST: SDD1

**Manual operation:** See ["Bit Frame SDD / Bit Frame Anticollision"](#) on page 52

---

**[ :SOURce<hw>]:BB:NFC:CBLOCK<ch>:BLOCK<st>:BDATa <BData>**

Sets the value of "Block Data" .

**Parameters:**

<BData> integer

**Manual operation:** See ["Block List, Block Data, Block List Configuration"](#) on page 53

---

**[ :SOURce<hw>]:BB:NFC:CBLOCK<ch>:BLKSelection <BlockSel>**

Selects a block to be read/written.

**Parameters:**

<BlockSel> integer  
 Range: 0 to 14  
 \*RST: 1

**Manual operation:** See ["Block or Byte Selection \(ADD\)"](#) on page 54

---

**[ :SOURce<hw>]:BB:NFC:CBLOCK<ch>:BLOCK<st>:BNUMber <BNumber>**

Sets the block number in the block list.

**Parameters:**

<BNumber> integer  
 Range: 0 to depends on block list length  
 \*RST: 0

**Manual operation:** See ["Block List, Block Data, Block List Configuration"](#) on page 53

---

**[ :SOURce<hw>]:BB:NFC:CBLOCK<ch>:BLOCK<st>:LEN <BLength>**

Sets the block length.

**Parameters:**

<BLength>            LEN2 | LEN3  
                          \*RST:        LEN2

**Manual operation:**    See ["Block List, Block Data, Block List Configuration"](#)  
                                  on page 53

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:BLOCK<st>:LOCKed <LControl>**

Enables/disables status information on lock for the corresponding block ("BLOCK-1" to "BLOCK-C").

**Parameters:**

<LControl>            1 | ON | 0 | OFF  
                          \*RST:        0

**Manual operation:**    See ["Lock Control or Status"](#) on page 59

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:BLOCK<st>:SLOrder <SCLOrder>**

Sets the service code list order.

**Parameters:**

<SCLOrder>            integer  
                          Range:        0 to dynamic  
                          \*RST:        0

**Manual operation:**    See ["Block List, Block Data, Block List Configuration"](#)  
                                  on page 53

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:BNO <BNo>**

Selects the block number to be read/write.

**Parameters:**

<BNo>                   integer  
                          Range:        0 to 255  
                          \*RST:        1

**Example:**                BB:NFC:CBLock<CH>:BNO 78  
                                  Selects the block number 78 to be read/written

**Manual operation:**    See ["Block Number \(BNo\)"](#) on page 54

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:BTYPe <BType>**

Selects the block type to be sent.

**Parameters:**

<BType>                TPI | TPR | TPS  
                          \*RST:        TPI

**Example:** See [Chapter 7.1, "Programming example"](#), on page 73.

**Manual operation:** See ["Block Type"](#) on page 54

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:BYTSelection <ByteSel>**

Selects a byte to be read/written.

**Parameters:**

<ByteSel> integer  
Range: 0 to 7  
\*RST: 1

**Manual operation:** See ["Block or Byte Selection \(ADD\)"](#) on page 54

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:CFGType <ConfType>**

Determines what platform or protocol the device in listen mode is configured for.

**Parameters:**

<ConfType> T2 | T4A | NDEP | DT4A | OFF | 0 | ON | 1  
\*RST: T2

**Manual operation:** See ["Configuration Type "](#) on page 54

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:CHaining <Chaining>**

Determines if chaining is applied.

**Parameters:**

<Chaining> 1 | ON | 0 | OFF  
\*RST: 0

**Manual operation:** See ["Chaining"](#) on page 54

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:CID <CID>**

Determines the value of CID.

**Parameters:**

<CID> float  
\*RST: 1

**Manual operation:** See ["DID \(DID field\)/ CID \(CID field\)"](#) on page 56

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:CI14 <ConfType>**

Determines what platform or protocol the device is configured for.

**Parameters:**

<ConfType> T2 | T4A | NDEP | DT4A | OFF | 0 | ON | 1

---

**[ :SOURce<hw>]:BB:NFC:CBLOCK<ch>:CSUPported <CSupported>**

Determines if CID is supported.

**Parameters:**

<CSupported>      0 | 1 | OFF | ON

**Manual operation:**    See ["DID Supported / CID Supported"](#) on page 56

---

**[ :SOURce<hw>]:BB:NFC:CBLOCK<ch>:DATA <Data>**

Selects the data source type.

**Parameters:**

<Data>              ZERO | ONE | PATtern | PN9 | PN11 | PN15 | PN16 | PN20 |  
PN21 | PN23 | DLISt  
\*RST:              PN9

**Manual operation:**    See ["Data List Management..."](#) on page 26  
See ["Data Source \(Data\)"](#) on page 55

---

**[ :SOURce<hw>]:BB:NFC:CBLOCK<ch>:DATA:DSElection <DSelection>**

Selects a Data List.

**Parameters:**

<DSelection>      string

**Manual operation:**    See ["Data List Management..."](#) on page 26  
See ["Data Source \(Data\)"](#) on page 55

---

**[ :SOURce<hw>]:BB:NFC:CBLOCK<ch>:DATA:LENGTH <Length>**

Determines the length of the transmitted user data / general data.

**Parameters:**

<Length>            integer  
Range:              0 to 65536  
\*RST:              0

**Example:**              See [Chapter 7.1, "Programming example"](#), on page 73

**Manual operation:**    See ["Data Length"](#) on page 55

---

**[ :SOURce<hw>]:BB:NFC:CBLOCK<ch>:DATA:PATtern <Pattern>, <BitCount>**

Defines a bit pattern.

**Parameters:**

<Pattern>            numeric  
\*RST:              #H0,1

<BitCount> integer  
 Range: 1 to 64  
 \*RST: 1

**Example:** See [Chapter 7.1, "Programming example"](#), on page 73.

**Manual operation:** See ["Data Source \(Data\)"](#) on page 55

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:DEQD <DivEqDiv>**

Determines if the same bit rate divisor for both directions is supported.

**Parameters:**

<DivEqDiv> 1 | ON | 0 | OFF  
 \*RST: 0

**Manual operation:** See ["D\(LISTEN->POLL\)=D\(POLL->LISTEN\) / D\(PICC->PCD\)=D\(PCD->PICC\)"](#) on page 57

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:DFOLLOWing <DFollowing>**

Determines if a DID is following.

**Parameters:**

<DFollowing> 1 | ON | 0 | OFF  
 \*RST: 0

**Manual operation:** See ["DID following \(I-block type, R-block type or S-block type\)"](#) on page 56

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:DID <DID>**

Determines the value of DID (Device Identification Number).

**Parameters:**

<DID> integer  
 Range: 0 to 14  
 \*RST: 1

**Manual operation:** See ["DID \(DID field\)/ CID \(CID field\)"](#) on page 56

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:DLP2 <TaDlp2>**

Enables support of divisor 2 for LISTEN to POLL (Bit Rate Capability).

**Parameters:**

<TaDlp2> 1 | ON | 0 | OFF  
 \*RST: 1

**Manual operation:** See ["D\(L->P=8\), D\(L->P=4\), D\(L->P=2\) "](#) on page 57



---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:DLP4 <TaDlp4>**

Enables support of divisor 4 for LISTEN to POLL (Bit Rate Capability).

**Parameters:**

<TaDlp4>            OFF | ON | 1 | 0  
                       \*RST:        ON

**Manual operation:**    See "[D\(L->P=8\)](#), [D\(L->P=4\)](#), [D\(L->P=2\)](#)" on page 57

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:DLP8 <TaDlp8>**

Enables support of divisor 8 for LISTEN to POLL (Bit Rate Capability).

**Parameters:**

<TaDlp8>            1 | ON | 0 | OFF  
                       \*RST:        1

**Manual operation:**    See "[D\(L->P=8\)](#), [D\(L->P=4\)](#), [D\(L->P=2\)](#)" on page 57

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:DLTPoll <Dltp>**

In ATTRIB command, sets the divisor in the corresponding transmission direction.

**Parameters:**

<Dltp>              DIV1 | DIV2 | DIV4 | DIV8  
                       \*RST:        DIV1

**Manual operation:**    See "[Divisor \(LISTEN to POLL\)](#), [Divisor \(POLL to LISTEN\)](#) / [Divisor \(PCD to PICC\)](#), [Divisor \(PICC to PCD\)](#)" on page 57

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:DPL2 <TaDpl2>**

Enables support of divisor 2 for POLL to LISTEN (Bit Rate Capability).

**Parameters:**

<TaDpl2>            1 | ON | 0 | OFF  
                       \*RST:        1

**Manual operation:**    See "[D\(P->L=8\)](#), [D\(P->L=4\)](#), [D\(P->L=2\)](#)" on page 57

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:DPL4 <TaDpl4>**

Enables support of divisor 4 for POLL to LISTEN (Bit Rate Capability).

**Parameters:**

<TaDpl4>            1 | ON | 0 | OFF  
                       \*RST:        1

**Manual operation:**    See "[D\(P->L=8\)](#), [D\(P->L=4\)](#), [D\(P->L=2\)](#)" on page 57

---

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:DPL8 <TaDpl83>**

Enables support of divisor 8 for POLL to LISTEN (Bit Rate Capability).

**Parameters:**

<TaDpl83>            1 | ON | 0 | OFF  
                       \*RST:        1

**Manual operation:**    See "[D\(P->L=8\)](#), [D\(P->L=4\)](#), [D\(P->L=2\)](#)" on page 57

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:DPPCd <DPPCd>**

In ATTRIB command, sets the divisor in the corresponding transmission direction.

**Parameters:**

<DPPCd>            DIV1 | DIV2 | DIV4 | DIV8

**Manual operation:**    See "[Divisor \(LISTEN to POLL\)](#), [Divisor \(POLL to LISTEN\)](#) / [Divisor \(PCD to PICC\)](#), [Divisor \(PICC to PCD\)](#)" on page 57

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:DPPicc <Dpp>**

In ATTRIB command, sets the divisor in the corresponding transmission direction.

**Parameters:**

<Dpp>            DIV1 | DIV2 | DIV4 | DIV8

**Manual operation:**    See "[Divisor \(LISTEN to POLL\)](#), [Divisor \(POLL to LISTEN\)](#) / [Divisor \(PCD to PICC\)](#), [Divisor \(PICC to PCD\)](#)" on page 57

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:DPTListen <Dptl>**

In ATTRIB command, sets the divisor in the corresponding transmission direction.

**Parameters:**

<Dptl>            DIV1 | DIV2 | DIV4 | DIV8  
                       \*RST:        DIV1

**Manual operation:**    See "[Divisor \(LISTEN to POLL\)](#), [Divisor \(POLL to LISTEN\)](#) / [Divisor \(PCD to PICC\)](#), [Divisor \(PICC to PCD\)](#)" on page 57

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:DRI <Dri>**

Sets DRI.

**Parameters:**

<Dri>            D1 | D2 | D8 | D4 | D16 | D32 | D64  
                       \*RST:        D1

**Manual operation:**    See "[DRI](#)" on page 58

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:DSI <Dsi>**

Sets DSI.

**Parameters:**

<Dsi>                    D1 | D2 | D8 | D4 | D16 | D32 | D64  
                           \*RST:        D1

**Manual operation:**    See ["DSI"](#) on page 58

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:DSUPported <DSupported>**

Determines if DID is supported.

**Parameters:**

<DSupported>        1 | ON | 0 | OFF  
                           \*RST:        0

**Manual operation:**    See ["DID Supported / CID Supported"](#) on page 56

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:DWSelection <DWSelection>**

Selects DESELECT or WTX.

**Parameters:**

<DWSelection>       DSEL | WTX  
                           \*RST:        DSEL

**Manual operation:**    See ["DESELECT or WTX"](#) on page 56

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:EASupported <EAtqb>**

Determines if Extended ATQB is supported.

**Parameters:**

<EAtqb>                0 | 1 | OFF | ON

**Manual operation:**    See ["Extended SENSB\\_RES / Extended ATQB"](#) on page 58

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:ESSupported <ESensbres>**

Determines if Extended SENSB\_RES is supported.

**Parameters:**

<ESensbres>        1 | ON | 0 | OFF  
                           \*RST:        0

**Manual operation:**    See ["Extended SENSB\\_RES / Extended ATQB"](#) on page 58

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:FSC <Fsc>**

Selects the maximum frame size in bytes.

**Parameters:**

<Fsc> F16 | F24 | F32 | F40 | F48 | F64 | F96 | F128 | F256  
 \*RST: F32

**Manual operation:** See ["FSC"](#) on page 58

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:FWI <Fwi>**

Determines the FWI which is needed to calculate Frame Waiting Time (FWT).

**Parameters:**

<Fwi> integer  
 Range: 1 to 8  
 \*RST: 4

**Manual operation:** See ["FWI"](#) on page 59

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:GBSelection <GBSelection>**

Selects 8-byte block to be read/written.

**Parameters:**

<GBSelection> integer  
 Range: 0 to 255  
 \*RST: 1

**Example:** BB:NFC:CBLock<CH>:GBSelection 122  
 Selects GB number 122.

**Manual operation:** See ["Global Block Selection \(ADD\)"](#) on page 59

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:GDAVailable <GDAVailable>**

Determines if General data is available.

**Parameters:**

<GDAVailable> 1 | ON | 0 | OFF  
 \*RST: 0

**Manual operation:** See ["General Data"](#) on page 59

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:IBNumber <IBNumber>**

Indicates if a Valid I-block or a Valid R(ACK) block is received.

**Parameters:**

<IBNumber> 1 | ON | 0 | OFF  
 \*RST: 0

**Manual operation:** See ["Block Number \(for I-block type or R-block type\)"](#) on page 54

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:KPARameter <KParameter>**

Determines the number of historical bytes (T1 to Tk).

**Parameters:**

<KParameter>	integer
Range:	0 to 15
*RST:	0

**Manual operation:** See "[k](#)" on page 59

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:LREDuction <LReduction>**

Selects the length reduction (LR).

**Parameters:**

<LReduction>	LR64   LR128   LR192   LR254
*RST:	LR64

**Manual operation:** See "[Length Reduction](#)" on page 59

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:MBLI <Mbli>**

Determines the Maximum Buffer Length Index (MBLI).

**Parameters:**

<Mbli>	integer
Range:	0 to 15
*RST:	1

**Manual operation:** See "[MBLI](#)" on page 59

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:MFSsize <MFSsize>**

Selects the maximum frame size in bytes.

**Parameters:**

<MFSsize>	F16   F24   F32   F40   F48   F64   F96   F128   F256
-----------	---

**Manual operation:** See "[FSC](#)" on page 58

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:MICHaining <MChaining>**

Determines if more information (MI) is chained.

**Parameters:**

<MChaining>	1   ON   0   OFF
*RST:	0

**Manual operation:** See "[MI \(more information\) Chaining](#)" on page 59

---

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:MTR0 <MTr0>**

Sets the minimum value of TR0 supported.

**Parameters:**

<MTr0>                      TR00 | TR01 | TR02  
                               \*RST:        TR00

**Manual operation:**    See "[Minimum TR0, TR1, TR2](#)" on page 60

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:MTR1 <MTr1>**

Sets the minimum value of TR1 supported.

**Parameters:**

<MTr1>                      TR10 | TR11 | TR12  
                               \*RST:        TR10

**Manual operation:**    See "[Minimum TR0, TR1, TR2](#)" on page 60

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:MTR2 <MTr2>**

Sets the minimum value of TR2 supported.

**Parameters:**

<MTr2>                      TR20 | TR21 | TR22 | TR23  
                               \*RST:        TR20

**Manual operation:**    See "[Minimum TR0, TR1, TR2](#)" on page 60

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:N2FType <NFType>**

Determines which protocol or platform the NFCID2 format is for.

**Parameters:**

<NFType>                    NDEP | TT3  
                               \*RST:        NDEP

**Manual operation:**    See "[NFCID2 Format Type](#)" on page 61

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:NACK <Nack>**

Determines the value of NACK.

**Parameters:**

<Nack>                      NCK1 | NCK0 | NCK4 | NCK5  
                               \*RST:        NCK0

**Manual operation:**    See "[NACK](#)" on page 60

---

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:NAD <Nad>**

Determines the value of NAD.

**Parameters:**

<Nad> integer

**Manual operation:** See ["NAD"](#) on page 60

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:NBLOCKS <NBlock>**

Determines the number of blocks.

**Parameters:**

<NBlock> integer  
 Range: 1 to dynamic  
 \*RST: 1

**Manual operation:** See ["Number of Blocks"](#) on page 62

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:NFOLLOWING <NFollowing>**

Determines if NAD is following.

**Parameters:**

<NFollowing> 1 | ON | 0 | OFF  
 \*RST: 0

**Manual operation:** See ["NAD following"](#) on page 60

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:NID0 <Nfcid0>**

Determines the entire value of NFCID0.

**Parameters:**

<Nfcid0> integer

**Manual operation:** See ["NFCID0 \(hex\) / PUPID \(hex\)"](#) on page 60

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:NID1 <Nfcid1>**

Determines the entire value of NFCID1.

**Parameters:**

<Nfcid1> integer

**Manual operation:** See ["NFCID1 \(hex\) / UID \(hex\)"](#) on page 60

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:NID2 <Nfcid2>**

Determines the entire value of NFCID2. Is a virtual parameter for SCPI to set the NFCID2 either in NFC-DEP or Type 3 Tag mode.

**Parameters:**

&lt;Nfcid2&gt; integer

**Manual operation:** See ["NFCID2 \(hex\)"](#) on page 61**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:NNComplete <Nfcid1NotCom>**

Determines whether NFCID1 is complete or not.

**Parameters:**

&lt;Nfcid1NotCom&gt; 1 | ON | 0 | OFF

\*RST: 0

**Manual operation:** See ["NFCID1 not complete / UID not complete"](#) on page 61**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:NOApplications <NOApplication>**

Determines the number of applications.

**Parameters:**

&lt;NOApplication&gt; integer

Range: 0 to 15

\*RST: 1

**Manual operation:** See ["Number of Applications"](#) on page 61**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:NOSlots <NOSlots>**

Determines the number of slots.

**Parameters:**

&lt;NOSlots&gt; S1 | S2 | S4 | S8 | S16

\*RST: S1

**Manual operation:** See ["Number of Slots"](#) on page 62**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:NSERVICES <NService>**

Sets the number of services.

**Parameters:**

&lt;NService&gt; integer

Range: 1 to 16

\*RST: 1

**Manual operation:** See ["Number of Services"](#) on page 62**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:NSIZE <Nfcid1sz>**

Determines the size of NFCID1.



**Parameters:**

<Nfcid1sz>            SINGLe | DOUBLe | TRIPlE  
                          \*RST:        SINGLe

**Manual operation:**    See ["NFCID1 Size / UID Size"](#) on page 61

**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:NSUPported <NSupport>**

Determines if NAD is supported.

**Parameters:**

<NSupport>            1 | ON | 0 | OFF  
                          \*RST:        0

**Manual operation:**    See ["NAD Supported"](#) on page 60

**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:PAD0 <Pad0>**

**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:PAD1 <Pad1>**

**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:PAD2 <Pad2>**

Sets the value of PAD0/PAD1/PAD2 (hex).

**Parameters:**

<Pad2>                integer  
                          \*RST:        #H00

**Manual operation:**    See ["PAD0, PAD1, PAD2"](#) on page 63

**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:PDUType <PDUType>**

Selects the type of PDU.

**Parameters:**

<PDUType>            INFO | ANACK | SUPer

**Manual operation:**    See ["PDU Type"](#) on page 63

**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:PLIN <PLIndication>**

Only used when DESELECT/WTX is set to WTX. Determines Power Level Indication.

**Parameters:**

<PLIndication>        integer  
                          Range:        0 to 3  
                          \*RST:        1

**Manual operation:**    See ["Power Level Indication"](#) on page 63

**[[:SOURce<hw>]:BB:NFC:CBLOCK<ch>:PLIR <PLIndicator>**

Sets the Power Level Indicator.

**Parameters:**

<PLIndicator> integer  
 Range: 0 to 3  
 \*RST: 1

**Manual operation:** See ["Power Level Indicator"](#) on page 64

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:PNI <Pni>**

Only used with PDU type Information. Determines Packet Number Information (PNI).

**Parameters:**

<Pni> integer  
 Range: 0 to 3  
 \*RST: 1

**Manual operation:** See ["PNI"](#) on page 63

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:PSElection <PSelection>**

Selects if the first or second packet of the SECTOR\_SELECT command is transmitted.

**Parameters:**

<PSelection> PCK1 | PCK2  
 \*RST: PCK1

**Manual operation:** See ["Packet Selection"](#) on page 63

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:PUPI <PUPI>**

Determines the entire value of PUPI.

**Parameters:**

<PUPI> integer

**Manual operation:** See ["NFCID0 \(hex\) / PUPI \(hex\)"](#) on page 60

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:RC <Rc>**

Indicates the Request Code (RC).

**Parameters:**

<Rc> NSCI | SCIR | APFS  
 \*RST: NSCI

**Manual operation:** See ["RC"](#) on page 64

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:RTOX <Rtox>**

Determines the response timeout extension request value (RTOX).

**Parameters:**

<Rtox> integer  
 Range: 1 to 59  
 \*RST: 1

**Manual operation:** See ["RTOX"](#) on page 64

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:SCMD <SCmd>**

Selects the cascade level (CL) of the NFCID1 requested by the NFC Forum Device in Poll Mode.

**Parameters:**

<SCmd> CL1 | CL2 | CL3  
 \*RST: CL1

**Manual operation:** See ["SEL\\_CMD / SEL"](#) on page 64

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:SCODE <SCode>**

Sets the System Code.

**Parameters:**

<SCode> integer  
 \*RST: #Hffff

**Manual operation:** See ["SC"](#) on page 64

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:SEGSelection <SegmentSel>**

Selects a segment to be read.

**Parameters:**

<SegmentSel> integer  
 Range: 0 to 15  
 \*RST: 1

**Manual operation:** See ["Segment Selection \(ADD\)"](#) on page 64

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:SEL <SElect>**

Selects the cascade level (CL) of the UID.

**Parameters:**

<SElect> CL1 | CL2 | CL3

**Manual operation:** See ["SEL\\_CMD / SEL"](#) on page 64

---

**[:SOURce<hw>]:BB:NFC:CBLock<ch>:SENRequired <SENRequired>**

Determines whether a suppression of EoS (End of Sequence)/SoS (Start of Sequence) is required or not.

**Parameters:**

<SENRequired>      1 | ON | 0 | OFF  
                          \*RST:      0

**Manual operation:**    See ["Suppression of EoS, SoS Not Required"](#) on page 66

---

**[:SOURce<hw>]:BB:NFC:CBLock<ch>:SERVICE<st>:AATtributes <AAttributes>**

Enables the Service Code List Configuration.

**Parameters:**

<AAttributes>          AARW | AARO  
                          \*RST:          AARW

**Manual operation:**    See ["Service Code List..."](#) on page 65

---

**[:SOURce<hw>]:BB:NFC:CBLock<ch>:SERVICE<st>:SNUMber <SNumber>**

Determines the number of services.

**Parameters:**

<SNumber>              integer  
                          Range:          0 to 1023  
                          \*RST:          0

**Manual operation:**    See ["Service Code List..."](#) on page 65

---

**[:SOURce<hw>]:BB:NFC:CBLock<ch>:SF1 <SFlag1>**

Sets the status flag 1 to specify a Type 3 tag's error condition. A value of 0 signals a successful execution, values different from 0 indicate errors.

**Parameters:**

<SFlag1>                integer

**Manual operation:**    See ["Status Flag 1, Status Flag 2"](#) on page 66

---

**[:SOURce<hw>]:BB:NFC:CBLock<ch>:SF2 <SFlag2>**

Sets the status flag 2 to specify a Type 3 tag's error condition. A value of 0 signals a successful execution, values different from 0 indicate errors.

**Parameters:**

<SFlag2>                integer

**Manual operation:**    See ["Status Flag 1, Status Flag 2"](#) on page 66

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:SFGI <Sfgi>**

Determines the Start-up Frame Guard Time (SFGT).

**Parameters:**

<Sfgi> integer  
 Range: 0 to 8  
 \*RST: 0

**Manual operation:** See "[SFGI](#)" on page 65

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:SNO <SNO>**

Only available when packet selection is set to Packet2. Determines the sector number.

**Parameters:**

<SNO> integer  
 Range: 0 to 254  
 \*RST: 1

**Manual operation:** See "[SNo](#)" on page 65

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:SNUMBER <SNumber>**

Determines the slot number.

**Parameters:**

<SNumber> SN2 | SN3 | SN4 | SN5 | SN6 | SN7 | SN8 | SN9 | SN10 | SN11 |  
 SN12 | SN13 | SN14 | SN15 | SN16  
 \*RST: SN2

**Manual operation:** See "[Slot Number](#)" on page 65

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:SPLower <SPLower>**

Determines the bit count.

**Parameters:**

<SPLower> integer  
 Range: 0 to 7  
 \*RST: 0

**Manual operation:** See "[SEL\\_PAR\\_LOWER](#)" on page 65

---

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:SPUPper <SPUpper>**

SEL\_PAR\_UPPER determines the number of full bytes of the SDD\_REQ part.

**Parameters:**

<SPUpper> integer  
 Range: 2 to 6  
 \*RST: 2

**Manual operation:** See ["SEL\\_PAR\\_UPPER"](#) on page 64

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:SSNRequired <SSNRequired>**

Determines whether a suppression of EoS (End of Sequence)/SoS (Start of Sequence) is required or not.

**Parameters:**

<SSNRequired> 1 | ON | 0 | OFF  
 \*RST: 0

**Manual operation:** See ["Suppression of EoS, SoS Not Required"](#) on page 66

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:T1TConfigured <T1TPConfigured>**

Determines whether Type 1 Tag platform is configured or not.

**Parameters:**

<T1TPConfigured> 1 | ON | 0 | OFF  
 \*RST: 0

**Manual operation:** See ["Type 1 Tag Platform Configured"](#) on page 66

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:T1TK <T1totk>**

For number of historical bytes k greater than 0: sets the historical bytes T1 to Tk.

**Parameters:**

<T1totk> integer

**Manual operation:** See ["T1 to Tk"](#) on page 66

**[[:SOURce<hw>]:BB:NFC:CBLock<ch>:TAIPicc <TNAIPicc>**

Sets the total number of applications in the PICC (Proximity Inductive Coupling Card), i.e. in the NFC Forum Device in listener mode.

**Parameters:**

<TNAIPicc> integer  
 Range: 0 to 15  
 \*RST: 1

**Manual operation:** See ["Total No. Apps in the PICC"](#) on page 66

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:TSN <Tsn>**

Indicates the TSN (Time Slot Number).

**Parameters:**

<Tsn>                      TSN1 | TSN2 | TSN4 | TSN8 | TSN16  
                              \*RST:        TSN4

**Manual operation:**    See ["Number of Time Slots"](#) on page 62

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:UID <UID>**

Determines the entire value of UID.

**Parameters:**

<UID>                      integer

**Manual operation:**    See ["NFCID1 \(hex\) / UID \(hex\)"](#) on page 60

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:UNComplete <UIDNotCom>**

Determines whether UID is complete or not.

**Parameters:**

<UIDCom>                      0 | 1 | OFF | ON

**Manual operation:**    See ["NFCID1 not complete / UID not complete"](#) on page 61

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:USIZe <UIDsz>**

Determines the size of UID.

**Parameters:**

<UIDsz>                      SINGLE | DOUBLE | TRIPLE

**Manual operation:**    See ["NFCID1 Size / UID Size"](#) on page 61

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:WT <Wt>**

Sets the Waiting Time (WT) that codes the Response Waiting Time (RWT).

**Parameters:**

<Wt>                          integer  
                                  Range:        0 to 8  
                                  \*RST:        1

**Manual operation:**    See ["WT"](#) on page 66

---

**[ :SOURce<hw>]:BB:NFC:CBLock<ch>:WTXM <Wtxm>**

Determines the WTXM. - Only used when DESELECT/WTX is set to WTX.

**Parameters:**

<Wtxm> integer  
 Range: 1 to 59  
 \*RST: 1

**Manual operation:** See ["WTXM \(INF field of S\(WTX\) request, response\)"](#) on page 66

## 7.7 Modulation settings

<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:MSET:BOUtput</a>	112
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:MSET:BRATe?</a>	112
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:MSET:SLOPe</a>	112
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:MSET:IMODulation</a>	113
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:MSET:MDEPth</a>	113
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:MSET:MINDex</a>	113
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:MSET:OSRise</a>	113
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:MSET:RCURve</a>	114
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:MSET:SRATe</a>	114
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:MSET:TFALI</a>	114
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:MSET:TLOW</a>	114
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:MSET:TRISe</a>	114
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:MSET:USFall</a>	115

---

### **[\[:SOURce<hw>\]:BB:NFC:MSET:BOUtput](#) <BOutput>**

When activated the signal at the baseband output changes between 0% and 100% voltage to be able to control the Reference Listeners.

**Parameters:**

<BOutput> 1 | ON | 0 | OFF  
 \*RST: 1

**Manual operation:** See ["Baseband Output"](#) on page 40

---

### **[\[:SOURce<hw>\]:BB:NFC:MSET:BRATe?](#)**

Returns the resulting bitrate for the current settings.

**Return values:**

<BRate> float

**Usage:** Query only

**Manual operation:** See ["Bit Rate"](#) on page 38

---

### **[\[:SOURce<hw>\]:BB:NFC:MSET:SLOPe](#) <ESlope>**

Determines the transition between the modulated and unmodulated parts (Edge/Slope).



**Parameters:**

<ESlope> 1 | ON | 0 | OFF  
 \*RST: 1

**Manual operation:** See ["Slope"](#) on page 39

**[ :SOURce<hw>]:BB:NFC:MSET:IMODulation <IModulation>**

When selected, inverse modulation will be used.

**Parameters:**

<IModulation> 1 | ON | 0 | OFF  
 \*RST: 0

**Manual operation:** See ["Inverse Modulation"](#) on page 39

**[ :SOURce<hw>]:BB:NFC:MSET:MDEPth <MDepth>**

Sets the modulation depth in %.

**Parameters:**

<MDepth> float  
 Range: 0 to 100  
 Increment: 0.01  
 \*RST: 100

**Manual operation:** See ["Modulation Depth"](#) on page 39

**[ :SOURce<hw>]:BB:NFC:MSET:MINdex <MIndex>**

Defines the signal's modulation index in %.

**Parameters:**

<MIndex> float  
 Range: 0 to 100  
 Increment: 0.01  
 \*RST: 12

**Manual operation:** See ["Modulation Index"](#) on page 38

**[ :SOURce<hw>]:BB:NFC:MSET:OSRise <ORise>**

Determines the size of the overshoot after the rising slope.

**Parameters:**

<ORise> float  
 Range: 0 to 42  
 Increment: 0.01  
 \*RST: 0

**Manual operation:** See ["Overshoot Rising Slope \(VOU\)"](#) on page 42

---

**[ :SOURce<hw>]:BB:NFC:MSET:RCURve <RCurve>**

When activated an "RLC curve" is applied to the signal, otherwise a linear ramp is used.

**Parameters:**

<RCurve>                    1 | ON | 0 | OFF  
                               \*RST:            1

**Manual operation:**    See ["RLC curve"](#) on page 41

---

**[ :SOURce<hw>]:BB:NFC:MSET:SRATe <SRate>**

Enters the sample rate, i.e. the time resolution of the generated signal.

**Parameters:**

<SRate>                    float  
                               Range:            depends on protocol mode to dynamic  
                               Increment:    0.001  
                               \*RST:            20E6

**Manual operation:**    See ["Sample Rate"](#) on page 40

---

**[ :SOURce<hw>]:BB:NFC:MSET:TFALI <TFall>**

Defines the fall time (90 to 5 %) in µs.

**Parameters:**

<TFall>                    float  
                               Range:            0 to dynamic  
                               Increment:    0.01  
                               \*RST:            1

**Manual operation:**    See ["Tfall 90-10%/Tfall 90-5% \(t1-t2\)"](#) on page 42

---

**[ :SOURce<hw>]:BB:NFC:MSET:TLOW <TLow>**

Defines the signals low time (below 5%) in µs.

**Parameters:**

<TLow>                    float  
                               Range:            0.4 to dynamic  
                               Increment:    0.01  
                               \*RST:            1.9

**Manual operation:**    See ["Tlow \(t2\)"](#) on page 39

---

**[ :SOURce<hw>]:BB:NFC:MSET:TRISe <TRise>**

Defines the signals rise time (5 to 90 %) in µs.

**Parameters:**

<TRise> float  
 Range: dynamic to dynamic  
 Increment: 0.01  
 \*RST: 0.6

**Example:** See [Chapter 7.1, "Programming example"](#), on page 73.

**Manual operation:** See ["Trise 10-90%/Trise 5-90% \(t3\)"](#) on page 42

**[ :SOURce<hw>]:BB:NFC:MSET:USFall <OFall>**

Determines the size of the undershoot (ringing) after the falling slope.

**Parameters:**

<OFall> float  
 Range: 0 to 42  
 Increment: 0.01  
 \*RST: 0

**Manual operation:** See ["Undershoot Falling Slope \(VOU\)"](#) on page 42

## 7.8 Clipping settings

[\[:SOURce<hw>\]:BB:NFC:CLIPping:LEVel.....](#) 115  
[\[:SOURce<hw>\]:BB:NFC:CLIPping:STATe.....](#) 115

**[ :SOURce<hw>]:BB:NFC:CLIPping:LEVel <Level>**

Sets the limit for clipping.

**Parameters:**

<Level> integer  
 Range: 1 to 100  
 \*RST: 100

**Manual operation:** See ["Clipping Level "](#) on page 67

**[ :SOURce<hw>]:BB:NFC:CLIPping:STATe <State>**

Switches baseband clipping on and off.

**Parameters:**

<State> 1 | ON | 0 | OFF  
 \*RST: 0

**Manual operation:** See ["Clipping State"](#) on page 67

## 7.9 Trigger settings

<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:ARM:EXECute</a> .....	116
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:EXECute</a> .....	116
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger[:EXTernal&lt;ch&gt;]:DELay</a> .....	116
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger[:EXTernal&lt;ch&gt;]:INHibit</a> .....	117
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:EXTernal:SYNChronize:OUTPut</a> .....	117
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:OBASeband:DELay</a> .....	117
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:OBASeband:DELay</a> .....	118
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:OBASeband:INHibit</a> .....	118
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:RMODE?</a> .....	118
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:SLENgth</a> .....	118
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:SLUNit</a> .....	119
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:SOURce</a> .....	119
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC[:TRIGger]:SEQUence</a> .....	119
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:TIME:DATE</a> .....	120
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:TIME:TIME</a> .....	120
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:TIME[:STATE]</a> .....	121

---

### **[\[:SOURce<hw>\]:BB:NFC:TRIGger:ARM:EXECute](#)**

Stops signal generation; a subsequent internal or external trigger event restarts signal generation.

**Usage:** Event

**Manual operation:** See ["Arm"](#) on page 30

---

### **[\[:SOURce<hw>\]:BB:NFC:TRIGger:EXECute](#)**

Executes trigger manually.

**Usage:** Event

**Manual operation:** See ["Execute Trigger"](#) on page 31

---

### **[\[:SOURce<hw>\]:BB:NFC:TRIGger\[:EXTernal<ch>\]:DELay](#) <Delay>**

Sets the trigger signal delay in samples on external triggering or on internal triggering via the second path (if applicable).

The numeric suffix to EXTernal distinguishes between the external trigger via the multi-purpose connectors.

**Parameters:**

<Delay>	float
Range:	0 to 65535
Increment:	0.01
*RST:	0

**Example:** BB:NFC:TRIG:SOUR EGT1  
sets an external trigger  
BB:NFC:TRIG:EXT:DEL 50  
sets a delay of 50 symbols for the trigger.

**Manual operation:** See ["External Inhibit/Trigger Inhibit"](#) on page 32

**[ :SOURce<hw>]:BB:NFC:TRIGger[:EXTernal<ch>]:INHibit <Inhibit>**

Sets the duration for inhibiting a new trigger event subsequent to triggering. The input is to be expressed in samples.

**Parameters:**

<Inhibit> integer  
Range: 0 to 67108863  
\*RST: 0

**Manual operation:** See ["External Delay/Trigger Delay"](#) on page 33

**[ :SOURce<hw>]:BB:NFC:TRIGger:EXTernal:SYNChronize:OUTPut <Output>**

Enables signal output synchronous to the trigger event.

**Parameters:**

<Output> 1 | ON | 0 | OFF  
\*RST: 1

**Manual operation:** See ["Sync. Output to External Trigger/Sync. Output to Trigger"](#) on page 31

**[ :SOURce<hw>]:BB:NFC:TRIGger:OBASeband:DELay <Delay>**

Specifies the trigger delay (expressed as a number of samples) for triggering by the trigger signal from the second path.

**Parameters:**

<Delay> float  
Range: 0 to 2147483647  
Increment: 0.01  
\*RST: 0

**Example:** BB:NFC:TRIG:SOUR INTB  
the internal trigger signal from the other path must be used  
BB:NFC:TRIG:OBAS:DEL 50  
sets a delay of 50 symbols for the trigger.

**Manual operation:** See ["External Inhibit/Trigger Inhibit"](#) on page 32

**[ :SOURce<hw>]:BB:NFC:TRIGger:OBASeband:DELay <Delay>**

Available for two-path instruments only. Internal other baseband trigger delay. Sets the trigger signal delay in samples on internal triggering via the second path.

**Parameters:**

<Delay> float  
 Range: 0 to 65535  
 Increment: 0.01  
 \*RST: 0

**Example:**

BB:NFC:TRIG:SOUR OBAS  
 sets for path A the internal trigger executed by the trigger signal from the second path (path B).  
 BB:NFC:TRIG:OBAS:DEL 50  
 sets a delay of 50 symbols for the trigger.

**[ :SOURce<hw>]:BB:NFC:TRIGger:OBASeband:INHibit <Inhibit>**

Available for two-path instruments only. Internal other baseband trigger inhibit. Sets the trigger signal inhibit in samples on internal triggering via the second path.

**Parameters:**

<Inhibit> integer  
 Range: 0 to 67108863  
 \*RST: 0

**Manual operation:** See ["External Delay/Trigger Delay"](#) on page 33

**[ :SOURce<hw>]:BB:NFC:TRIGger:RMODE?**

Queries the status of signal generation.

**Return values:**

<RMode> STOP | RUN

**Usage:** Query only

**Manual operation:** See ["Running/Stopped"](#) on page 30

**[ :SOURce<hw>]:BB:NFC:TRIGger:SLENgth <Slength>**

Defines the length of the signal sequence to be output in the SINGLe trigger mode.

**Parameters:**

<Slength> integer  
 Range: 1 to 4294967295.0  
 \*RST: 1

**Manual operation:** See ["Signal Duration"](#) on page 30

---

**[[:SOURce<hw>]:BB:NFC:TRIGger:SLUNit <Slunit>**

Defines the unit for the entry of the signal sequence length.

**Parameters:**

<Slunit>                      SEQuence | SAMPlE  
                                  \*RST:           SEQuence

**Manual operation:**    See ["Signal Duration Unit"](#) on page 29

---

**[[:SOURce<hw>]:BB:NFC: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
  - ELTRigger: External local trigger
  - ELClock: External local clock
- Internal triggering by a signal from the other basebands (INTA | INTB)
- 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 SMW accepts these values and maps them automatically as follows:  
 EXTernal = EGT1, BEXTernal = EGT2, OBASeband = INTA or INTB  
 (depending on the current baseband)

**Parameters:**

<Source>                      INTB|INTernal|OBASeband|EGT1|EGT2|EGC1|EGC2|ELTRigger|INTA|ELClock|BEXTernal|EXTernal | BBSY  
                                  \*RST:           INTernal

**Example:**                      BB:NFC:TRIG:SOUR INT  
                                  Selects an internal trigger source.

**Options:**                      ELTRigger|ELClock require R&S SMW-B10  
                                  BBSY require R&S SMW-B9

**Manual operation:**    See ["Source"](#) on page 31

---

**[[:SOURce<hw>]:BB:NFC[:TRIGger]:SEQuence <Sequence>**

Selects a regular trigger mode.

**Parameters:**

<Sequence>            AUTO | RETRigger | AAUTO | ARETrigger | SINGLE  
 \*RST:                AUTO

**Manual operation:**    See ["Mode"](#) on page 29

**[[:SOURce<hw>]:BB:NFC: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>                 integer  
                          Range:        1 to 31

**Example:**            See example "Configure a time-based trigger signal" in the subchapter "Trigger Commands" of the chapter "SOURce:BB:ARB subsystem" in the R&S SMW user manual.

**Manual operation:**    See ["Trigger Time"](#) on page 30

**[[:SOURce<hw>]:BB:NFC:TRIGger:TIME:TIME <Hour>, <Minute>, <Second>**

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:**

<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 SMW user manual.

**Manual operation:** See ["Trigger Time"](#) on page 30

---

**[SOURce<hw>]:BB:NFC:TRIGger:TIME[:STATe] <State>**

Activates time-based triggering with a fixed time reference. If activated, the R&S SMW 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:**

<State> 1 | ON | 0 | OFF

**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 SMW user manual.

**Manual operation:** See ["Time Based Trigger"](#) on page 30

## 7.10 Marker settings

<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:OUTPut&lt;ch&gt;:MODE</a>	121
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:OUTPut&lt;ch&gt;:ONTime</a>	122
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:OUTPut&lt;ch&gt;:OFFTime</a>	122
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:OUTPut&lt;ch&gt;:PATtern</a>	122
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:OUTPut&lt;ch&gt;:PULSe:DIVider</a>	122
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:OUTPut&lt;ch&gt;:PULSe:FREQuency?</a>	123
<a href="#">[:SOURce&lt;hw&gt;]:BB:NFC:TRIGger:OUTPut&lt;ch&gt;:DELay</a>	123

---

**[SOURce<hw>]:BB:NFC:TRIGger:OUTPut<ch>:MODE <Mode>**

**Parameters:**

<Mode> PULSe | REStart | PATtern | RATio

**PULSe**

A regular marker signal is generated. The frequency is derived by dividing the sample rate by the divider, which is input with the command [\[:SOURce<hw>\]:BB:NFC:TRIGger:OUTPut<ch>:PULSe:DIVider](#) on page 122.

**REStart**

A marker signal is generated on every repetition of the complete frame sequence.

**PATtern**

A marker signal that is defined by a bit pattern is generated. The pattern has a maximum length of 64 bits and is defined with the command `[ :SOURce<hw> ] :BB:NFC:TRIGger:OUTPut<ch>:PATtern` on page 122

**RATio**

A marker signal corresponding to the Time Off / Time On specifications in the commands `[ :SOURce<hw> ] :BB:NFC:TRIGger:OUTPut<ch>:ONTime` on page 122 and `[ :SOURce<hw> ] :BB:NFC:TRIGger:OUTPut<ch>:OFFTime` on page 122

\*RST: REStart

**Manual operation:** See "Mode" on page 34

`[ :SOURce<hw> ] :BB:NFC:TRIGger:OUTPut<ch>:ONTime <OnTime>`  
`[ :SOURce<hw> ] :BB:NFC:TRIGger:OUTPut<ch>:OFFTime <OffTime>`

Sets the number of samples in a period (ON time + time, during which the marker signal on the marker output is OFF) for marker `RATio`.

**Parameters:**

<OffTime> integer  
 Range: 1 to 16777215  
 \*RST: 1

**Manual operation:** See "Mode" on page 34

`[ :SOURce<hw> ] :BB:NFC:TRIGger:OUTPut<ch>:PATtern <Pattern>, <BitCount>`

A marker signal that is defined by a bit pattern is generated.

**Parameters:**

<Pattern> numeric  
 \*RST: #H2  
 <BitCount> integer  
 Range: 1 to 64  
 \*RST: 2

**Manual operation:** See "Mode" on page 34

`[ :SOURce<hw> ] :BB:NFC:TRIGger:OUTPut<ch>:PULSe:DIVider <Divider>`

Sets the divider for the pulsed marker signal.

**Parameters:**

<Divider> integer  
 Range: 2 to 1024  
 \*RST: 2

**Example:** BB:NFC:TRIG:OUTP2:PULS:DIV 2

**Manual operation:** See ["Mode"](#) on page 34

**[ :SOURce<hw> ]:BB:NFC:TRIGger:OUTPut<ch>:PULSe:FREQuency?**

Queries the pulse frequency of the pulsed marker signal.

**Return values:**

<Frequency> float  
Range: 0.0 to max

**Example:** BB:NFC:TRIG:OUTP2:MODE PULS  
BB:NFC:TRIG:OUTP2:PULS:DIV 4  
BB:NFC:TRIG:OUTP2:PULS:FREQ?  
Response: 600.000 Hz

**Usage:** Query only

**Manual operation:** See ["Mode"](#) on page 34

**[ :SOURce<hw> ]:BB:NFC:TRIGger:OUTPut<ch>:DELay <Delay>**

Defines the delay between the signal on the marker outputs and the start of the signal, expressed in terms of the signal units.

**Parameters:**

<Delay> float  
Range: 0 to 16777215  
Increment: 0.001  
\*RST: 0  
Default unit: Samples

**Manual operation:** See ["Delay"](#) on page 35

## 7.11 Clock settings

[\[ :SOURce<hw> \]:BB:NFC:CLOCK:MODE](#)..... 123

[\[ :SOURce<hw> \]:BB:NFC:CLOCK:SOURce](#)..... 124

**[ :SOURce<hw> ]:BB:NFC:CLOCK:MODE <Mode>**

Sets the type of externally supplied clock.

**Parameters:**

<Mode> MSAMple | SAMPlE  
\*RST: SAMPlE

**Manual operation:** See ["Clock Mode"](#) on page 36

---

**[[:SOURce<hw>]:BB:NFC: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|ELCLock|EXTernal  
\*RST:            INTernal

**Example:**            BB:NFC:CLOC:SOUR INT  
Selects an internal clock reference.

**Options:**            ELCLock requires R&S SMW-B10

**Manual operation:** See "[Clock Source](#)" on page 36

# Glossary: List of terms and abbreviations

## A

**Active Communication:** A communication mode in which each device generates its own RF field to send a message to another device.

**Activity:** A process within an NFC Forum Device with well defined pre-conditions and post-conditions, as defined in [ACTIVITY]. An Activity can only start when its pre-conditions are fulfilled. When an Activity ends, its post-conditions are fulfilled.

**AID:** Application ID

**ASK:** Amplitude Shift Keying

**ATN:** Attention.

## B

**BCC:** UID CLn check byte for NFC-A

**bd:** Bit Duration

**BPSK:** Bi Phase Shift Keying

## C

**Card Emulator:** A role of an NFC Forum Device, reached when an NFC Forum Device in Listen Mode has gone through a number of Activities and in which the NFC Forum Device behaves as one of the Technology Subsets.

**CID:** Cryptogram Information Data

**CLn:** Cascade Level n ( $1 \leq n \leq 3$ )

**Command:** An instruction from one device to another device in order to move the other device through a state machine.

**Connectionless Transport:** An unacknowledged data transmission service with minimal protocol complexity.

**Correct Frame:** A frame without Transmission Error.

**CRC:** Cyclic Redundancy Check, a checksum appended within the data segment before transmission, and verified afterwards by the recipient to detect transmission errors

**CRC\_A:** CRC error detection code for NFC-A.

**CRC\_B:** CRC error detection code for NFC-B.

**CRC\_F:** CRC error detection code for NFC-F.

**CT:** Command type.

## D

**DID:** Device Identification Number.

**DRI:** Codes the bit rate in communication direction from Target to Initiator.

**DSI:** Codes the bit rate in communication direction from Initiator to Target.

## E

**EoD:** End of Data.

**EoF:** End of Frame.

**EoS:** End of sequence.

## F

**fc:** Carrier frequency.

**FSC:** Maximum frame size (in bytes).

**FWI:** Frame waiting time integer.

**FWT:** Frame waiting time.

## I

**IC:** Integrated Circuit

**Initiator:** A role of an NFC Forum Device reached when an NFC Forum Device in Poll Mode has gone through a number of Activities; in this mode the NFC Forum Device communicates using the NFC-DEP Protocol.

**ISO-DEP Protocol:** Half-duplex block transmission protocol defined in Section 13 and based on [ISO/IEC\_14443] and [EMV\_CLESS].

## L

**Listen Frame:** A frame sent by an NFC Forum Device in Listen Mode.

**Listen Mode:** Initial mode of an NFC Forum Device when it does not generate a carrier; in this mode the NFC Forum Device listens for the RF field of another device.

**lsb:** least significant bit

**LSB:** Least Significant Byte

## M

**MBL:** Maximum Buffer Length

**MBLI:** Maximum Buffer Length Index

**MRT:** Maximum response time

**MRTI:** Maximum response time information

**msb:** Most Significant Bit

**MSB:** Most Significant Byte

## N

**NAD:** Node Addressing

**NDEF:** NFC data exchange format.

**NFC:** Near Field Communication

**NFC Forum Device:** A device that supports the following Modus Operandi: Initiator, Target, and Reader/Writer. It may also support Card Emulator.

**NFC Tag:** A contactless tag or (smart) card supporting NDEF over Passive Communication.

**NFC-A:** Near Field Communication - Type A Technology

**NFC-B:** Near Field Communication - Type B Technology

**NFC-DEP Protocol:** Half-duplex block transmission protocol defined in Section 14 and based on [ISO/IEC\_18092].

**NFC-F:** Near Field Communication - Type F Technology

**NFCID0:** NFC-B identifier of the NFC Forum Device.

**NFCID1:** NFC-A identifier of the NFC Forum Device in the passive communication mode.

**NFCID2:** NFC-F identifier of the NFC Forum Device in the passive communication mode.

**NFCID3:** NFCIP-1 identifier of the NFC Forum Device. NFCID3 is always 10 byte long.

**NFCIP-1:** Near field communication interface and protocol as specified in [ISO/IEC\_18092].

**NRZ-L:** Non-Return to Zero (L for Level)

## O

**OOK:** On-Off Keying

**Operating Field:** The magnetic field created by an NFC Forum Device in poll mode within the operating volume.

**Operating Volume:** The three-dimensional space, as defined by the NFC Forum, in which an NFC Forum Device in Poll Mode can communicate with an NFC Forum Device in Listen Mode.

## P

**Passive Communication:** A communication mode in which one device generates an RF field and sends Commands to a second device. To respond, this second device uses load modulation (i.e., it does not generate an RF field but it draws more or less power from the RF field).

**PCB:** Protocol Control Byte.

**PCD:** Proximity Coupling Device.

**PDU:** Protocol Data Unit.

**PICC:** Proximity Inductive Coupling Card

**Poll Command:** A Command to query an NFC Forum Device in Listen Mode or an NFC Forum Tag:

- ALL\_REQ or SENS\_REQ Command for NFC-A
- ALLB\_REQ or SENSB\_REQ Command for NFC-B
- SENSEF\_REQ Command for NFC-F

**Poll Frame:** A frame sent by an NFC Forum Device in Poll Mode.

**Poll Mode:** Initial mode of an NFC Forum Device when it generates a carrier and probes ("polls") for other devices.

**Protocol Error:** A Semantic Error or Syntax Error.



**PUPI:** Pseudo-Unique PICC Identifier available for EMV Type B.

## R

**Reader/Writer:** Role of an NFC Forum Device reached when an NFC Forum Device in Poll Mode has gone through a number of Activities. In this mode, the NFC Forum Device behaves like a legacy contactless reader and uses Commands from one of the Technology Subsets.

**Response:** Information sent from one device to another device upon receipt of a Command. The information received by the other device should allow this other device to continue the data exchange.

**RRDD:** Reader-Reader Data Delay

**RWT:** Response Waiting Time

## S

**SDD:** Single Device Detection

**Semantic Error:** A Correct Frame with no Syntax Error is received when it is not expected.

**SFGI:** Start-up Frame Guard Time Integer.

**SFGT:** Start-up Frame Guard Time

**SoD:** Start of Data

**SoF:** Start of Frame

**SoS:** Start of Sequence

**Syntax Error:** A Correct Frame is received with an invalid content. In this case, the coding of the Command or the block within the frame is not consistent with this specification.

## T

**Target:** Role of an NFC Forum Device, reached when the NFC Forum Device has gone through a number of Activities in which the NFC Forum Device communicates using the NFC-DEP Protocol.

**Technology:** A group of transmission parameters defined by the NFC standard that make a complete communication protocol. A non-exhaustive list of transmission parameters is: RF carrier, communication mode, bit rate, modulation scheme, bit level coding, frame format, protocol, and Command set. NFC defines three groups and therefore three Technologies: NFC-A, NFC-B, and NFC-F. The three Technologies use

the same RF carrier (13.56 MHz). Each Technology uses its own modulation scheme, bit level coding and frame format, but may have the same protocol and Command set.

**Technology Subset:** A legacy platform supporting a subset of a Technology. A Technology Subset supports at least the Poll Command of the Technology. The four Technology Subsets described in the NFC Digital Protocol Technical Specification are:

- Type 1 Tag platform, which uses a particular subset of NFC-A, excluding anti-collision.
- Type 2 Tag platform, which uses a particular subset of NFC-A, including anti-collision.
- Type 3 Tag platform, which uses a particular subset of NFC-F, including anti-collision.
- Type 4 Tag platform, which uses a particular subset of NFC-A or NFC-B, including anti-collision.

**Timeout Error:** No Response has been received within the Response Waiting Time (RWT).

**Transmission Error:** An incorrect frame is received. In this case, the signal modulation, the bit coding, the frame format, the timing, or the checksum is not consistent with this specification.

## U

**UID:** Unique IDentifier available for EMV Type A.

## V

**Valid Block, Valid PDU:** A block or PDU without Protocol Error within a Correct Frame.

**Valid Command, Valid Response:** A Command or Response without Protocol Error within a Correct Frame.

## W

**WT:** Waiting Time, parameter to code RWT

**WTX:** Waiting Time Extension, containing 1 byte long INF field.

## List of commands

[SOURce<hw>]:BB:NFC:CBLock:APPend.....	81
[SOURce<hw>]:BB:NFC:CBLock<ch>:ACHK.....	87
[SOURce<hw>]:BB:NFC:CBLock<ch>:ADATa.....	87
[SOURce<hw>]:BB:NFC:CBLock<ch>:ADCoding.....	87
[SOURce<hw>]:BB:NFC:CBLock<ch>:AFI.....	88
[SOURce<hw>]:BB:NFC:CBLock<ch>:AID.....	88
[SOURce<hw>]:BB:NFC:CBLock<ch>:ALEngth.....	88
[SOURce<hw>]:BB:NFC:CBLock<ch>:ANSelection.....	88
[SOURce<hw>]:BB:NFC:CBLock<ch>:APFSupported.....	88
[SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:BOData.....	89
[SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:BOLEngth.....	89
[SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:FTYPE.....	89
[SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:SHData.....	89
[SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:SHLEngth.....	90
[SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:STData.....	90
[SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:STDLengTh?.....	90
[SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:STEPresent.....	91
[SOURce<hw>]:BB:NFC:CBLock<ch>:APGeneric:STPLEngth.....	91
[SOURce<hw>]:BB:NFC:CBLock<ch>:ATIMEout.....	91
[SOURce<hw>]:BB:NFC:CBLock<ch>:AUPD.....	87
[SOURce<hw>]:BB:NFC:CBLock<ch>:BCCError.....	91
[SOURce<hw>]:BB:NFC:CBLock<ch>:BCHK.....	87
[SOURce<hw>]:BB:NFC:CBLock<ch>:BFANTicollision.....	91
[SOURce<hw>]:BB:NFC:CBLock<ch>:BFSDd.....	92
[SOURce<hw>]:BB:NFC:CBLock<ch>:BLKSelection.....	92
[SOURce<hw>]:BB:NFC:CBLock<ch>:BLOCK<st>:BDATa.....	92
[SOURce<hw>]:BB:NFC:CBLock<ch>:BLOCK<st>:BNUmber.....	92
[SOURce<hw>]:BB:NFC:CBLock<ch>:BLOCK<st>:LEn.....	92
[SOURce<hw>]:BB:NFC:CBLock<ch>:BLOCK<st>:LOCKed.....	93
[SOURce<hw>]:BB:NFC:CBLock<ch>:BLOCK<st>:SLORder.....	93
[SOURce<hw>]:BB:NFC:CBLock<ch>:BNO.....	93
[SOURce<hw>]:BB:NFC:CBLock<ch>:BPGeneric:DATA.....	90
[SOURce<hw>]:BB:NFC:CBLock<ch>:BPGeneric:DLEngTh?.....	90
[SOURce<hw>]:BB:NFC:CBLock<ch>:BPGeneric:PLEngTh.....	91
[SOURce<hw>]:BB:NFC:CBLock<ch>:BTYPe.....	93
[SOURce<hw>]:BB:NFC:CBLock<ch>:BUPD.....	87
[SOURce<hw>]:BB:NFC:CBLock<ch>:BYTSelection.....	94
[SOURce<hw>]:BB:NFC:CBLock<ch>:CFGType.....	94
[SOURce<hw>]:BB:NFC:CBLock<ch>:CHAining.....	94
[SOURce<hw>]:BB:NFC:CBLock<ch>:CID.....	94
[SOURce<hw>]:BB:NFC:CBLock<ch>:CI14.....	94
[SOURce<hw>]:BB:NFC:CBLock<ch>:COPY.....	81
[SOURce<hw>]:BB:NFC:CBLock<ch>:CSUPported.....	95
[SOURce<hw>]:BB:NFC:CBLock<ch>:CTYPe.....	82
[SOURce<hw>]:BB:NFC:CBLock<ch>:DATA.....	95
[SOURce<hw>]:BB:NFC:CBLock<ch>:DATA:DSElection.....	95
[SOURce<hw>]:BB:NFC:CBLock<ch>:DATA:LENGTh.....	95

[SOURce<hw>]:BB:NFC:CBLock<ch>:DATA:PATtern.....	95
[SOURce<hw>]:BB:NFC:CBLock<ch>:DELeTe.....	82
[SOURce<hw>]:BB:NFC:CBLock<ch>:DEQD.....	96
[SOURce<hw>]:BB:NFC:CBLock<ch>:DFOLlowing.....	96
[SOURce<hw>]:BB:NFC:CBLock<ch>:DID.....	96
[SOURce<hw>]:BB:NFC:CBLock<ch>:DLP2.....	96
[SOURce<hw>]:BB:NFC:CBLock<ch>:DLP4.....	97
[SOURce<hw>]:BB:NFC:CBLock<ch>:DLP8.....	97
[SOURce<hw>]:BB:NFC:CBLock<ch>:DLTPoll.....	97
[SOURce<hw>]:BB:NFC:CBLock<ch>:DPL2.....	97
[SOURce<hw>]:BB:NFC:CBLock<ch>:DPL4.....	97
[SOURce<hw>]:BB:NFC:CBLock<ch>:DPL8.....	98
[SOURce<hw>]:BB:NFC:CBLock<ch>:DPPCd.....	98
[SOURce<hw>]:BB:NFC:CBLock<ch>:DPPicc.....	98
[SOURce<hw>]:BB:NFC:CBLock<ch>:DPTListen.....	98
[SOURce<hw>]:BB:NFC:CBLock<ch>:DRI.....	98
[SOURce<hw>]:BB:NFC:CBLock<ch>:DSI.....	99
[SOURce<hw>]:BB:NFC:CBLock<ch>:DSUPported.....	99
[SOURce<hw>]:BB:NFC:CBLock<ch>:DURation.....	82
[SOURce<hw>]:BB:NFC:CBLock<ch>:DWSeLection.....	99
[SOURce<hw>]:BB:NFC:CBLock<ch>:EASupported.....	99
[SOURce<hw>]:BB:NFC:CBLock<ch>:ECHK.....	87
[SOURce<hw>]:BB:NFC:CBLock<ch>:ESSupported.....	99
[SOURce<hw>]:BB:NFC:CBLock<ch>:EUPD.....	87
[SOURce<hw>]:BB:NFC:CBLock<ch>:FPGeneric:DATA.....	90
[SOURce<hw>]:BB:NFC:CBLock<ch>:FPGeneric:DLEnGth?.....	90
[SOURce<hw>]:BB:NFC:CBLock<ch>:FPGeneric:PLEnGth.....	91
[SOURce<hw>]:BB:NFC:CBLock<ch>:FSC.....	99
[SOURce<hw>]:BB:NFC:CBLock<ch>:FWI.....	100
[SOURce<hw>]:BB:NFC:CBLock<ch>:GBSeLection.....	100
[SOURce<hw>]:BB:NFC:CBLock<ch>:GDAVailable.....	100
[SOURce<hw>]:BB:NFC:CBLock<ch>:IBNumber.....	100
[SOURce<hw>]:BB:NFC:CBLock<ch>:INSert.....	82
[SOURce<hw>]:BB:NFC:CBLock<ch>:KPARameter.....	101
[SOURce<hw>]:BB:NFC:CBLock<ch>:LREDuction.....	101
[SOURce<hw>]:BB:NFC:CBLock<ch>:MBLI.....	101
[SOURce<hw>]:BB:NFC:CBLock<ch>:MFSize.....	101
[SOURce<hw>]:BB:NFC:CBLock<ch>:MICHaining.....	101
[SOURce<hw>]:BB:NFC:CBLock<ch>:MTR0.....	102
[SOURce<hw>]:BB:NFC:CBLock<ch>:MTR1.....	102
[SOURce<hw>]:BB:NFC:CBLock<ch>:MTR2.....	102
[SOURce<hw>]:BB:NFC:CBLock<ch>:N2FType.....	102
[SOURce<hw>]:BB:NFC:CBLock<ch>:NACK.....	102
[SOURce<hw>]:BB:NFC:CBLock<ch>:NAD.....	103
[SOURce<hw>]:BB:NFC:CBLock<ch>:NBLOCKS.....	103
[SOURce<hw>]:BB:NFC:CBLock<ch>:NFOLlowing.....	103
[SOURce<hw>]:BB:NFC:CBLock<ch>:NID0.....	103
[SOURce<hw>]:BB:NFC:CBLock<ch>:NID1.....	103
[SOURce<hw>]:BB:NFC:CBLock<ch>:NID2.....	103
[SOURce<hw>]:BB:NFC:CBLock<ch>:NNComplete.....	104

[SOURce<hw>]:BB:NFC:CBLock<ch>:NOApplications.....	104
[SOURce<hw>]:BB:NFC:CBLock<ch>:NOSlots.....	104
[SOURce<hw>]:BB:NFC:CBLock<ch>:NSERVICES.....	104
[SOURce<hw>]:BB:NFC:CBLock<ch>:NSIZE.....	104
[SOURce<hw>]:BB:NFC:CBLock<ch>:NSUPPORTED.....	105
[SOURce<hw>]:BB:NFC:CBLock<ch>:PAD0.....	105
[SOURce<hw>]:BB:NFC:CBLock<ch>:PAD1.....	105
[SOURce<hw>]:BB:NFC:CBLock<ch>:PAD2.....	105
[SOURce<hw>]:BB:NFC:CBLock<ch>:PASTE.....	83
[SOURce<hw>]:BB:NFC:CBLock<ch>:PDUTYPE.....	105
[SOURce<hw>]:BB:NFC:CBLock<ch>:PLIN.....	105
[SOURce<hw>]:BB:NFC:CBLock<ch>:PLIR.....	105
[SOURce<hw>]:BB:NFC:CBLock<ch>:PNI.....	106
[SOURce<hw>]:BB:NFC:CBLock<ch>:POFFset.....	83
[SOURce<hw>]:BB:NFC:CBLock<ch>:PSELECTION.....	106
[SOURce<hw>]:BB:NFC:CBLock<ch>:PUPIL.....	106
[SOURce<hw>]:BB:NFC:CBLock<ch>:RC.....	106
[SOURce<hw>]:BB:NFC:CBLock<ch>:REPETITION.....	83
[SOURce<hw>]:BB:NFC:CBLock<ch>:RTOX.....	106
[SOURce<hw>]:BB:NFC:CBLock<ch>:SAMPLES?.....	83
[SOURce<hw>]:BB:NFC:CBLock<ch>:SCMD.....	107
[SOURce<hw>]:BB:NFC:CBLock<ch>:SCODE.....	107
[SOURce<hw>]:BB:NFC:CBLock<ch>:SEGSELECTION.....	107
[SOURce<hw>]:BB:NFC:CBLock<ch>:SEL.....	107
[SOURce<hw>]:BB:NFC:CBLock<ch>:SENREQUIRED.....	108
[SOURce<hw>]:BB:NFC:CBLock<ch>:SERVICE<st>:AATTRIBUTES.....	108
[SOURce<hw>]:BB:NFC:CBLock<ch>:SERVICE<st>:SNUMBER.....	108
[SOURce<hw>]:BB:NFC:CBLock<ch>:SF1.....	108
[SOURce<hw>]:BB:NFC:CBLock<ch>:SF2.....	108
[SOURce<hw>]:BB:NFC:CBLock<ch>:SFGI.....	109
[SOURce<hw>]:BB:NFC:CBLock<ch>:SNO.....	109
[SOURce<hw>]:BB:NFC:CBLock<ch>:SNUMBER.....	109
[SOURce<hw>]:BB:NFC:CBLock<ch>:SPLower.....	109
[SOURce<hw>]:BB:NFC:CBLock<ch>:SPUPper.....	109
[SOURce<hw>]:BB:NFC:CBLock<ch>:SSNREQUIRED.....	110
[SOURce<hw>]:BB:NFC:CBLock<ch>:STIME?.....	84
[SOURce<hw>]:BB:NFC:CBLock<ch>:T1TConfigured.....	110
[SOURce<hw>]:BB:NFC:CBLock<ch>:T1TK.....	110
[SOURce<hw>]:BB:NFC:CBLock<ch>:TAIPicc.....	110
[SOURce<hw>]:BB:NFC:CBLock<ch>:TSN.....	111
[SOURce<hw>]:BB:NFC:CBLock<ch>:UID.....	111
[SOURce<hw>]:BB:NFC:CBLock<ch>:UNComplete.....	111
[SOURce<hw>]:BB:NFC:CBLock<ch>:USIZE.....	111
[SOURce<hw>]:BB:NFC:CBLock<ch>:WT.....	111
[SOURce<hw>]:BB:NFC:CBLock<ch>:WTXM.....	111
[SOURce<hw>]:BB:NFC:CCBlock.....	81
[SOURce<hw>]:BB:NFC:CLIPPING:LEVEL.....	115
[SOURce<hw>]:BB:NFC:CLIPPING:STATE.....	115
[SOURce<hw>]:BB:NFC:CLOCK:MODE.....	123
[SOURce<hw>]:BB:NFC:CLOCK:SOURce.....	124

[SOURce<hw>]:BB:NFC:DCBLock.....	82
[SOURce<hw>]:BB:NFC:DIVisor.....	76
[SOURce<hw>]:BB:NFC:DVOLtage.....	76
[SOURce<hw>]:BB:NFC:ICBLock.....	82
[SOURce<hw>]:BB:NFC:MSET:BOUtput.....	112
[SOURce<hw>]:BB:NFC:MSET:BRATe?.....	112
[SOURce<hw>]:BB:NFC:MSET:IMODulation.....	113
[SOURce<hw>]:BB:NFC:MSET:MDEPth.....	113
[SOURce<hw>]:BB:NFC:MSET:MINdex.....	113
[SOURce<hw>]:BB:NFC:MSET:OSRIse.....	113
[SOURce<hw>]:BB:NFC:MSET:RCURve.....	114
[SOURce<hw>]:BB:NFC:MSET:SLOPe.....	112
[SOURce<hw>]:BB:NFC:MSET:SRATe.....	114
[SOURce<hw>]:BB:NFC:MSET:TFALI.....	114
[SOURce<hw>]:BB:NFC:MSET:TLOW.....	114
[SOURce<hw>]:BB:NFC:MSET:TRISe.....	114
[SOURce<hw>]:BB:NFC:MSET:USFall.....	115
[SOURce<hw>]:BB:NFC:PCBLock.....	83
[SOURce<hw>]:BB:NFC:PRED:APPLy.....	80
[SOURce<hw>]:BB:NFC:PRED:CNFMarker.....	80
[SOURce<hw>]:BB:NFC:PRED:SEQuence.....	80
[SOURce<hw>]:BB:NFC:PRESet.....	77
[SOURce<hw>]:BB:NFC:SCONfiguration:TNSamples?.....	84
[SOURce<hw>]:BB:NFC:SCONfiguration:TSDuration?.....	84
[SOURce<hw>]:BB:NFC:SETTing:CATalog?.....	79
[SOURce<hw>]:BB:NFC:SETTing:DELeTe.....	79
[SOURce<hw>]:BB:NFC:SETTing:LOAD.....	79
[SOURce<hw>]:BB:NFC:SETTing:STORe.....	79
[SOURce<hw>]:BB:NFC:STATe.....	77
[SOURce<hw>]:BB:NFC:TECHnology.....	77
[SOURce<hw>]:BB:NFC:TMODE.....	77
[SOURce<hw>]:BB:NFC:TRIGger:ARM:EXECute.....	116
[SOURce<hw>]:BB:NFC:TRIGger:EXECute.....	116
[SOURce<hw>]:BB:NFC:TRIGger:EXTernal:SYNChronize:OUTPut.....	117
[SOURce<hw>]:BB:NFC:TRIGger:OBASeband:DELay.....	117
[SOURce<hw>]:BB:NFC:TRIGger:OBASeband:DELay.....	118
[SOURce<hw>]:BB:NFC:TRIGger:OBASeband:INHibit.....	118
[SOURce<hw>]:BB:NFC:TRIGger:OUTPut<ch>:DELay.....	123
[SOURce<hw>]:BB:NFC:TRIGger:OUTPut<ch>:MODE.....	121
[SOURce<hw>]:BB:NFC:TRIGger:OUTPut<ch>:OFFTime.....	122
[SOURce<hw>]:BB:NFC:TRIGger:OUTPut<ch>:ONTime.....	122
[SOURce<hw>]:BB:NFC:TRIGger:OUTPut<ch>:PATtern.....	122
[SOURce<hw>]:BB:NFC:TRIGger:OUTPut<ch>:PULSe:DIVider.....	122
[SOURce<hw>]:BB:NFC:TRIGger:OUTPut<ch>:PULSe:FREQuency?.....	123
[SOURce<hw>]:BB:NFC:TRIGger:RMODE?.....	118
[SOURce<hw>]:BB:NFC:TRIGger:SLENgth.....	118
[SOURce<hw>]:BB:NFC:TRIGger:SLUNit.....	119
[SOURce<hw>]:BB:NFC:TRIGger:SOURce.....	119
[SOURce<hw>]:BB:NFC:TRIGger:TIME:DATE.....	120
[SOURce<hw>]:BB:NFC:TRIGger:TIME:TIME.....	120

[SOURce<hw>]:BB:NFC:TRIGger:TIME[:STATe].....	121
[SOURce<hw>]:BB:NFC:TRIGger[:EXternal<ch>]:DELay.....	116
[SOURce<hw>]:BB:NFC:TRIGger[:EXternal<ch>]:INHibit.....	117
[SOURce<hw>]:BB:NFC:UAISetting.....	77
[SOURce<hw>]:BB:NFC:UPVoltage?.....	78
[SOURce<hw>]:BB:NFC:VERSion?.....	78
[SOURce<hw>]:BB:NFC:WAVeform:CREate.....	78
[SOURce<hw>]:BB:NFC[:TRIGger]:SEQuence.....	119

# Index

## A

Analog I/Q settings for desired voltage .....	27
Append a command block .....	49
Application cards .....	10
Application notes .....	10
Apply .....	43
Arm Trigger .....	116
Armed_Auto .....	29
Armed_Retrigger .....	29
Auto .....	29

## B

Baseband clipping .....	67
Brochures .....	9

## C

Chip clock .....	36
Clipping	
level .....	67
state .....	67
Clock	
Mode .....	36
Source .....	36
Command block configuration .....	49
Command type .....	45
Command type overview .....	46
Common trigger settings .....	29
Configure marker 1 for the sequence .....	43
Configure the peak voltage in the analog I/Q out block ....	27
Conventions	
SCPI commands .....	72
Copy and paste a command block .....	49
Coupled trigger settings .....	29
Create and edit a data list .....	26

## D

Data list .....	26
Data sheets .....	9
Default settings .....	25
Delay	
Marker .....	35
Delete a command block .....	49
Desired voltage .....	27
Disable EMV .....	24
Disable NFC .....	24
Documentation overview .....	8

## E

Edit a data list .....	26
Edit the sequence configuration .....	49
EMV type .....	26
Enable EMV .....	24
Enable NFC .....	24
External clock type .....	123

## F

Frame configuration .....	49
Frame period determination .....	49

## G

Generate waveform file .....	26
Getting started .....	8

## H

Help .....	8
------------	---

## I

Insert a command block .....	49
Installation .....	7
Instrument help .....	8
Instrument security procedures .....	9

## L

Leveling aspects .....	20
Listen mode .....	26
Load settings .....	25

## M

Manual trigger .....	31
Marker	
Period .....	34
Marker delay .....	35
Measured external clock .....	36

## N

NFC	
State .....	24
NFC mode .....	26
NFC-F	
data rate setting .....	26
divisor .....	26
Number of samples in the current frame .....	49

## O

Offset .....	49
On/Off ratio marker .....	34
ON/OFF Ratio Marker .....	122
Open source acknowledgment (OSA) .....	9
Overview: command types with SCPI-command parameters .....	46

## P

PCD to PICC mode .....	26
Peak voltage configuration in the analog I/Q out block ....	27
PICC to PCD mode .....	26
Poll mode .....	26
Power offset .....	49
Predefined sequence .....	43
Preset .....	25
Protocol mode .....	26
Protocol modes and corresponding command types - over- view .....	46

## R

Release notes .....	9
---------------------	---



Reset .....	25
Retrigger .....	29
RF level .....	20
RMS signal level .....	20
Running .....	30

## S

Safety instructions .....	9
Save NFC settings .....	25
SCPI-command parameters for command types .....	46
Security procedures .....	9
Sequence duration .....	44
Service manual .....	9
Standard settings .....	25
State	
NFC .....	24
Stopped .....	30

## T

Time-based trigger	
Date .....	30
State .....	30
Time .....	30
Transmission mode .....	26
Trigger	
Mode .....	29
Source .....	31
Tutorials .....	9

## U

Unmodulated parts voltage to peak voltage ratio .....	27
Unmodulated signal parts voltage .....	27
Update analog I/Q settings .....	27
User manual .....	8

## V

Videos .....	10
--------------	----

## W

White papers .....	10
--------------------	----