Remote Emulation with the R&S®SMC100A Signal Generator
Application Note

Products:

| R&S®SMC100A

The R&S®SMC100A signal generator offers a remote emulation feature that makes it possible to control the instrument by commands other than the built-in native SCPI commands. This feature allows the user to replace signal generators, e.g. from other manufacturers, with the R&S®SMC100A without having to change the remote control code.

This application note describes how to use the remote emulation feature in general. Furthermore, it describes in detail the remote emulation for each supported instrument, limitations of the individual emulations and the remaining differences between the emulated and the original commands.

ROHDE&SCHWARZ

Table of Contents

1 Abbreviations

The following abbreviations are used in this application note:

- AF2023   2023 signal generator from Aeroflex / IFR / Marconi
- AF2024   2024 signal generator from Aeroflex / IFR / Marconi
- AF2030   2030 signal generator from Aeroflex / IFR / Marconi
- AF2031   2031 signal generator from Aeroflex / IFR / Marconi
- AF2032   2032 signal generator from Aeroflex / IFR / Marconi
- AF2040   2040 signal generator from Aeroflex / IFR / Marconi
- AF2041   2041 signal generator from Aeroflex / IFR / Marconi
- AF2042   2042 signal generator from Aeroflex / IFR / Marconi
- E4428   E4428 signal generator from Agilent Technologies
- N5161   N5161 signal generator from Agilent Technologies
- N5181   N5181 signal generator from Agilent Technologies
- HP8642   8642 signal generator from Hewlett-Packard / Agilent Technologies
- HP8643   8643 signal generator from Hewlett-Packard / Agilent Technologies
- HP8644   8644 signal generator from Hewlett-Packard / Agilent Technologies
- HP8645   8645 signal generator from Hewlett-Packard / Agilent Technologies
- HP8647   8647 signal generator from Hewlett-Packard / Agilent Technologies
- HP8648   8648 signal generator from Hewlett-Packard / Agilent Technologies
- HP8656   8656 signal generator from Hewlett-Packard / Agilent Technologies
- HP8657   8657 signal generator from Hewlett-Packard / Agilent Technologies
- HP8664   8664 signal generator from Hewlett-Packard / Agilent Technologies
- HP8665   8665 signal generator from Hewlett-Packard / Agilent Technologies
- PA8303   8303 signal generator from Panasonic
- SMC   R&S$^{®}$SMC100A signal generator from Rohde & Schwarz
- SME   R&S$^{®}$SML signal generator from Rohde & Schwarz
- SML   R&S$^{®}$SML signal generator from Rohde & Schwarz
- SMT   R&S$^{®}$SMT signal generator from Rohde & Schwarz
- SMY   R&S$^{®}$SMY signal generator from Rohde & Schwarz

2 Overview

Measuring instruments used in commercial test systems for applications such as testing of mobile radio base stations typically have a much longer lifespan than the test system itself. Some five to seven years of usage is normal prior to replacement. On the other hand, test systems for use in aerospace & defense applications can have a typical lifespan of 25 years or even more. If standard measuring instruments with a lifespan that is significantly less than this time frame are used in such test systems, an obvious question is how to handle maintenance, repair, calibration and even replacement of the instruments that are no longer supported by their manufacturer. Moreover, the test program sets used in such test systems are generally certified, which makes it very time-consuming and costly to modify and reaccept them.

A feasible strategy to solve this problem is the migration with new measuring instruments that emulate the discontinued instruments.

This migration approach is saves time and is cost-effective. It involves replacing obsolete measuring instruments with new ones that emulate their functionality and programming.

The requirements for the new instruments revolve around the electrical and functional features of the instruments to be emulated as well as the test programs, which ideally should not require any modifications.

To fulfill the last requirement, the SMC offers the remote emulation feature.

Remote emulation means that the SMC will understand the programming commands of the emulated instrument and also emulate the behavior as fully as possible.

Most instruments from other manufacturers in the category of the SMC are emulated by the SMC. An overview of actually implemented remote emulations is given in the following table:

Remote emulations in the SMC			
Manufacturer	**Instrument**	**Language**	**Section**
Aeroflex IFR Marconi	AF2023	SCPI	5
	AF2024		
	AF2030	SCPI	6
	AF2031		
	AF2032		
	AF2040		
	AF2041		
	AF2042		
Agilent Technologies Hewlett-Packard	E4428	SCPI	7
	N5161		

| Remote emulations in the SMC			
Manufacturer	**Instrument**	**Language**	**Section**
Agilent Technologies Hewlett-Packard	N5181		
	HP8642	Non-SCPI	8
	HP8643	SCPI	9
	HP8644		
	HP8645		
	HP8647	SCPI	10
	HP8648		
	HP8656	Non-SCPI	11
	HP8657		
	HP8664	SCPI	9
	HP8665		
Panasonic	PA8303	Non-SCPI	12
Rohde & Schwarz	SML01	SCPI	13
	SML02		
	SML03		
	SME02	SCPI	14
	SME03		
	SME06		
	SMT02	SCPI	14
	SMT03		
	SMT04		
	SMY01	SCPI	15
	SMY02		

3 Basics

3.1 Remote Control Languages

Instruments implement different kind of remote control languages. These languages are grouped into two basic categories:

- SCPI-compatible
- Non-SCPI-compatible

Command examples				
Function	**HP8642** **Non-SCPI**	**HP8657** **Non-SCPI**	**AF2023** **SCPI**	**SMC** **SCPI**
Resetting the instrument	PR	PR	*RST	*RST
Clearing the system status		CS	*CLS	*CLS
Setting the RF frequency	CW10MZ	FR10MZ	:CFRQ:VALUE 10MHZ	:FREQ 10MHZ
Setting the RF power	AP-10DB	PL-10DM	:RFLV:VALUE -10DBM	:POW -10DBM
Activating the RF output	ON	R3	:RFLV:ON	:OUTP 1

Older instruments often implement a simple, unstructured and non-SCPI-compatible language, whereas modern instruments implement usually a complex, well structured and SCPI-compatible language.
The SMC offers a solution for both kinds of languages using specialized processing units in non-SCPI-compatible languages:
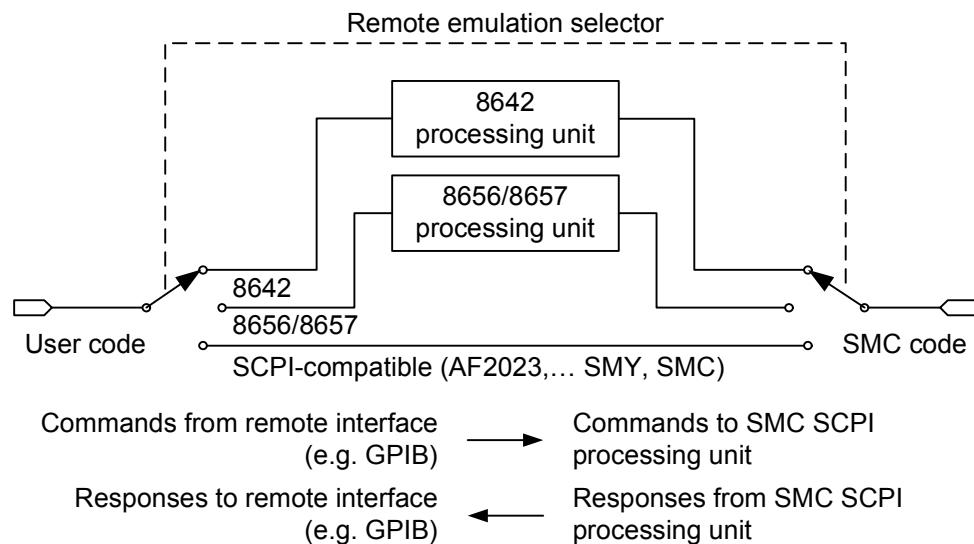
3.1.1 SCPI-Compatible Languages

Commands are routed directly from the remote interface to the SMC SCPI command processing unit; responses are routed in the reverse direction.

3.1.2 Non-SCPI-Compatible Languages

Commands are routed from the remote interface to the SMC SCPI command processing unit through a remote emulation specific processing unit; responses are routed in the reverse direction.
This specific processing unit parses the non-SCPI-compatible commands with reference to the corresponding syntax and translates them into SCPI-compatible ones. The unit also reformats the responses with respect to the requirements of the emulated instrument.

3.2 Remote Emulation Compatibility

An emulated instrument having fewer features than, or the same features as, the SMC can be replaced without special care.
However, replacing an emulated instrument having more features than the SMC or features that differ from those of the SMC requires additional care. The user must

- ensure that the SMC complies with the functional requirements
- verify that application code does not use features in the emulated instrument which are not available in the SMC

3.2.1 Command Compatibility

Most of the remote emulations in the SMC implement all commands of the original instrument. However, e.g. due to functional differences in hardware, in certain remote emulations the SMC

- does not support all commands
- does not support all parameters of a command
- implements a different behavior for a command

In the command tables of the different remote emulations, the status column gives compatibility information to a command:

Command compatibility status	
Status	**Comment**
✓	Command implementation is fully compatible.

Command compatibility status	
Status	**Comment**
☝ See item n	Command implementation is not fully compatible. The implementation in the SMC • does not support the same parameter(s) as the emulated instrument does • has different functionality than the emulated instrument • reports an invalid parameter or execution error if possible
○	Command is implemented without any functionality. The implementation in the SMC • ignores setting commands • returns default value in query commands • does not report errors • does not change any operating mode of the instrument • does not change any system state of the instrument
✘	Command is not implemented. The implementation in the SMC reports an unknown command error if possible.
+	Command has been added to enhance the functionality of the emulated instrument.

If the application software uses commands that are fully compatible, no special care has to be taken. The applications software can be used as is.
If the application software uses commands that are not fully compatible, the application software must be verified and normally also modified. If the required modifications to the application software are infeasible, the SMC cannot be used as replacement for another instrument.

3.2.2 IDN / OPT Strings

The remote emulation provides user-defined responses to *IDN? and *OPT? queries. This feature is of informational character only and has no impact on the functionality of the SMC.

3.2.3 Timing

The remote emulation cannot provide exact timing compatibility with the emulated instrument, since timing is a hardware-related property.

3.2.4 Sweep Operation

In the different sweep operating modes, often the behavior when a sweep has finished can be configured.
In some of the emulated instruments, the corresponding signal can be configured to remain at the stop value (upper figure) or to retrace to the start value (lower figure):
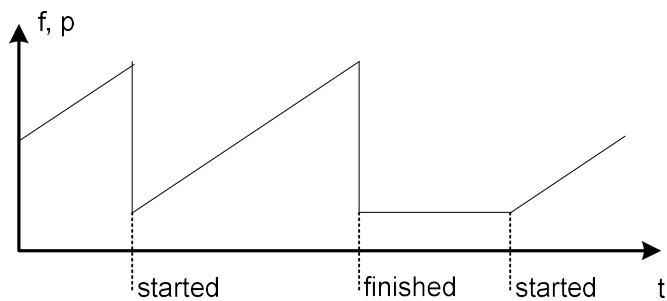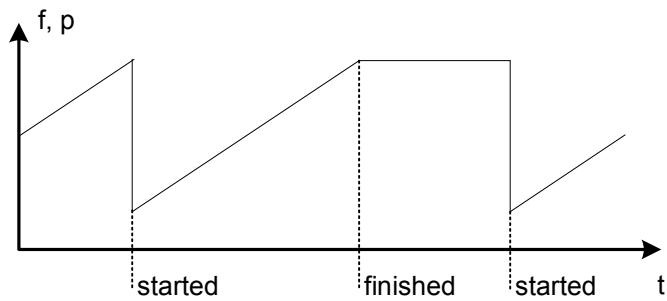
In the SMC, however, the corresponding signal always remains at the stop value:
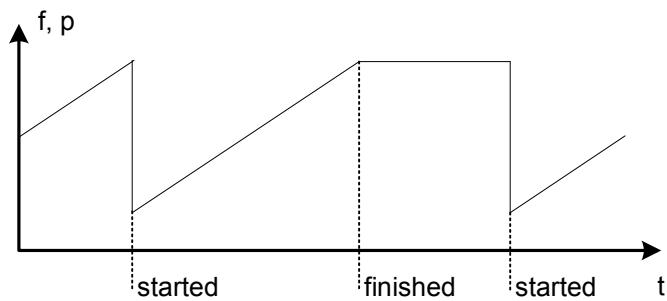
If this feature is undesirable, the user application code has to be changed.

Some of the emulated instruments implement sweep operations with a constant total sweep time:
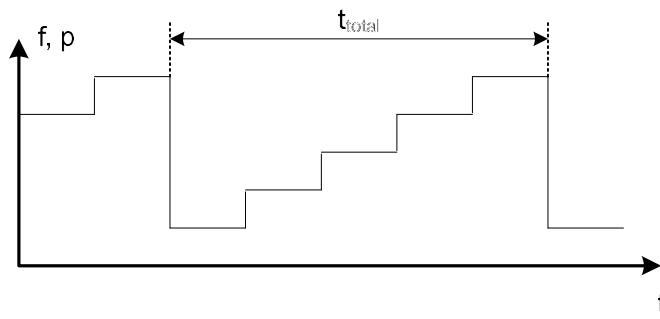
Changing the start, stop or steps parameter of the sweep operation does not have an impact on the total sweep time.

In the SMC, all sweep operations are based on the dwell time:
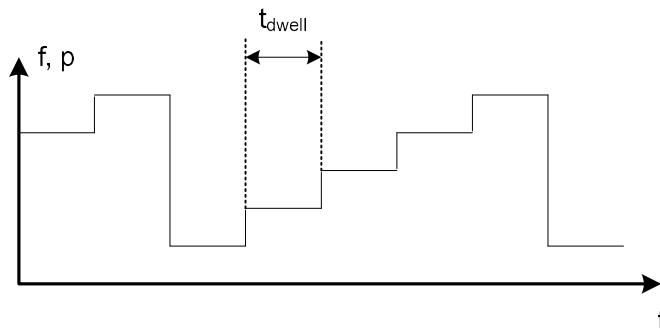
Changing the start, stop or steps parameter directly affects the total sweep time:

$$n_{steps} = \left| \frac{f_{stop} - f_{start}}{f_{step}} \right| + 1$$

$$t_{total} = n_{stels} \cdot t_{dwell}$$

3.2.5 Trigger Control

In signal generators with sweeping features, all sweep operating modes use the trigger control system. The figure below shows the SCPI trigger control system proposal:

pon or *RST or :ABOR

Idle

trigger system initiated
(:INIT[:IMM] or :INIT:CONT ON)

no longer initiated
(:INIT:CONT OFF)

Initiated

still initiated

number of arm
loops completed

Wait for arm

Wait for arm

Wait for arm

arm events

Wait for arm

arm conditions satisfied

number of trigger
loops completed

trigger event

Wait for trigger

trigger conditions satisfied

„device action" completed

Device action
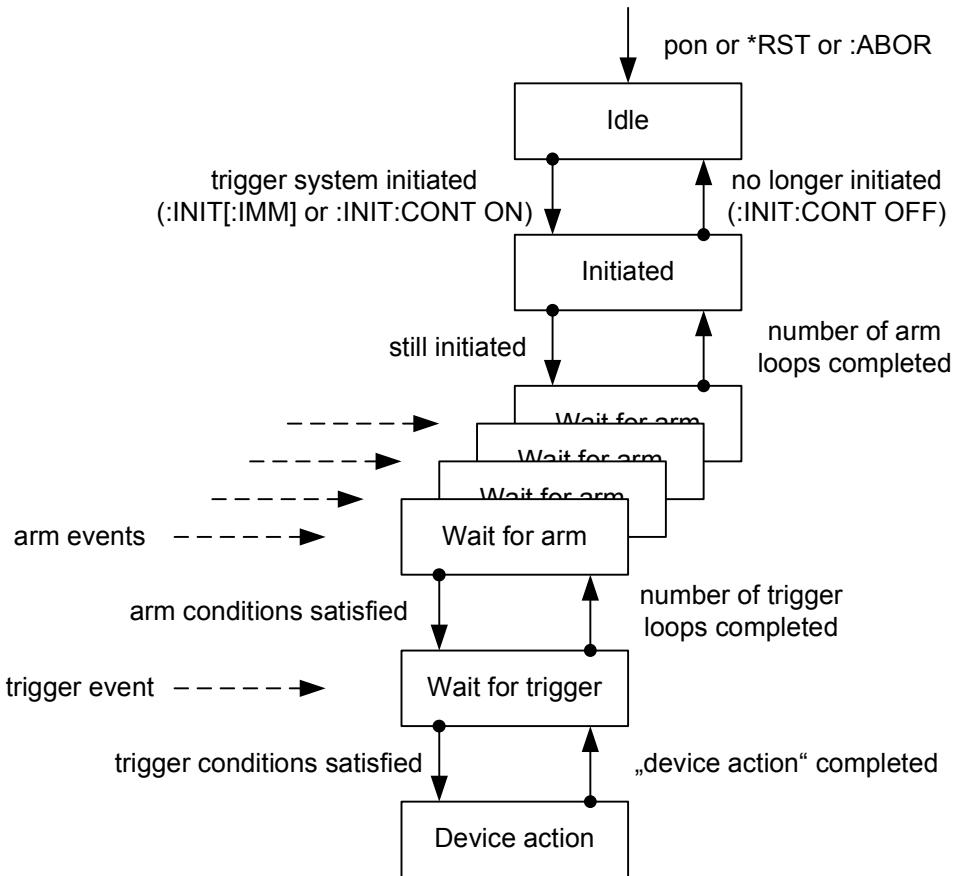
Some of the emulated instruments implement a simplified trigger control system:

pon or *RST or :ABOR

Idle

trigger system initiated
(:INIT[:IMM] or :INIT:CONT ON)

no longer initiated
(:INIT:CONT OFF)

Initiated

still initiated

number of trigger
loops completed

trigger event

Wait for trigger

trigger conditions satisfied

„device action"
completed

Device action
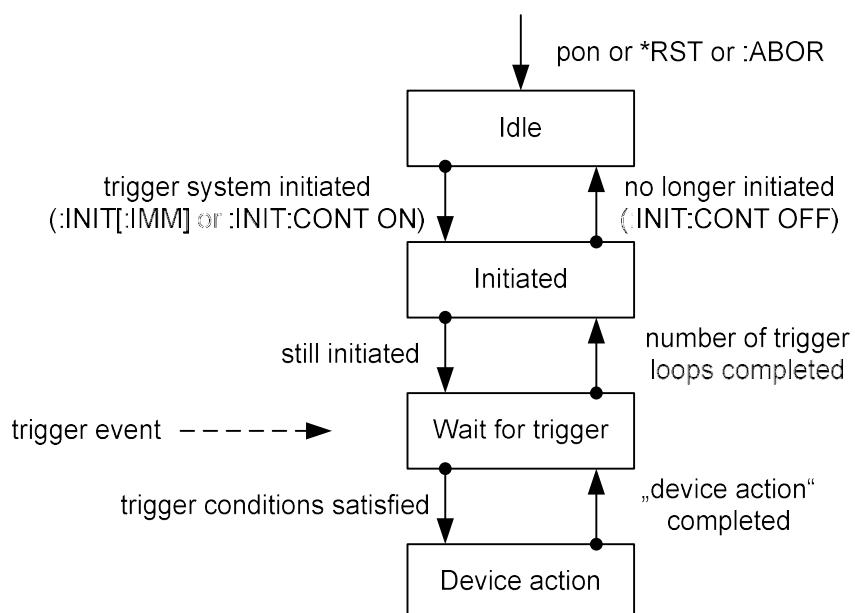
Other emulated instruments and also the SMC implement the trigger control system shown in the following figure:

```
                              │  pon or *RST or :ABOR
                              ▼
                    ┌───────────────────┐
                    │       Idle        │
                    └───────────────────┘
                              ●
trigger system initiated      │
                              ▼
                    ┌───────────────────┐
trigger event  ─ ─ ─ ─ ─ ─►   │  Wait for trigger  │
                    └───────────────────┘
                              ●         ▲
trigger conditions satisfied  │         │   „device action"
                              ▼         ●      completed
                    ┌───────────────────┐
                    │   Device action   │
                    └───────────────────┘
```
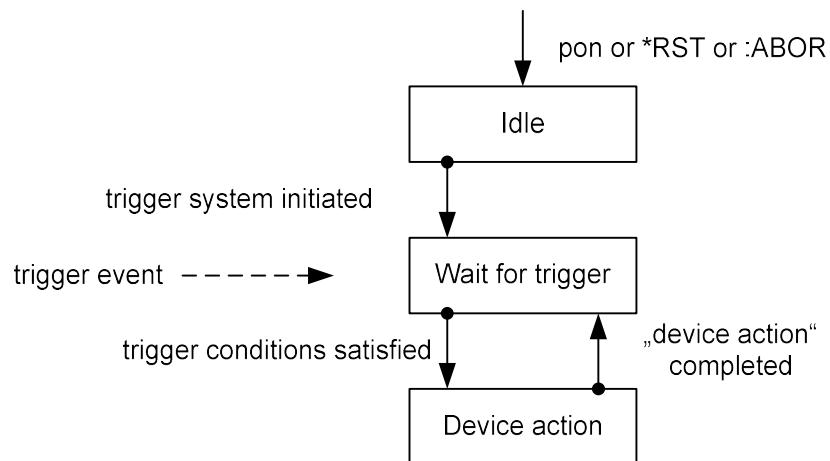
The main difference between the emulated instrument's and the SMC's trigger control system is that there is no "Initiated" state in the SMC. The SMC implementation assumes that the trigger system is initiated automatically in the "Idle" state. As a result, any sweep operation in the SMC, once activated, can only be controlled by internal or external trigger events.

This has an important impact on the user application. Application code that requires an "Initiated" state in the trigger control system must be adapted.

3.2.6 Status and Error Reporting

The SMC implements the minimal status and error reporting system required by the SCPI proposal. The following figure shows the status and error reporting model:
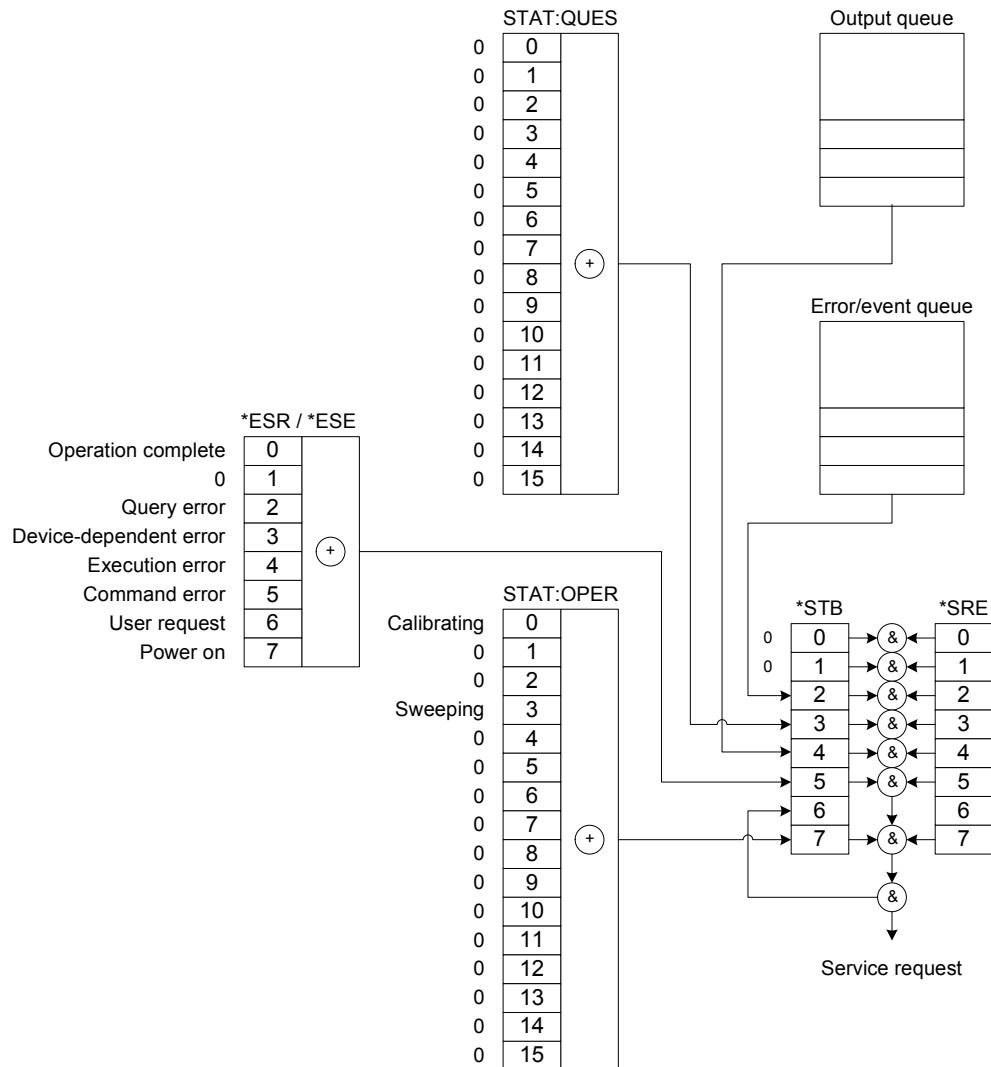
Some of the emulated instruments implement a more detailed status and error reporting system. Since the additional information stored in that system is not available in the SMC, application code that uses the additional information must be changed.

3.3 Preset / Reset

Changing the remote emulation does not automatically trigger a reset operation to the instrument. Therefore it is strongly recommended to manually execute a reset to the SMC after changing the remote emulation.
To apply the default of a particular remote emulation, the user must send the corresponding command via the remote control interface, e.g. the *RST command in SCPI-compatible languages.

3.4 Power Down / Power Up

The selected remote emulation and the user-defined responses to *IDN? and *OPT? queries are saved when the instrument is switched off.
When the instrument is switched on again, it starts up with the same settings that were active before it was switched off.

4 Activating a Remote Emulation

In order to use a specific remote emulation, it must first be activated by the user. Activation is done either

– manually using the SMC front panel
– remotely using SCPI commands

4.1 Manual Operation

On the SMC front panel, press the SETUP key to open the Menu tree:
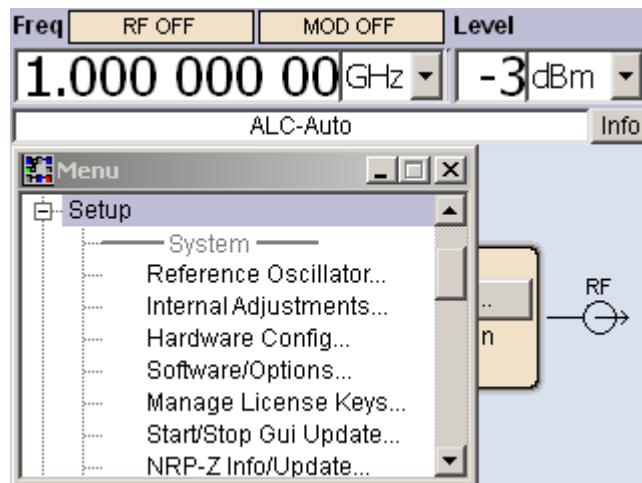
In the Menu tree, select GPIB… and open the Remote Channel Settings dialog:
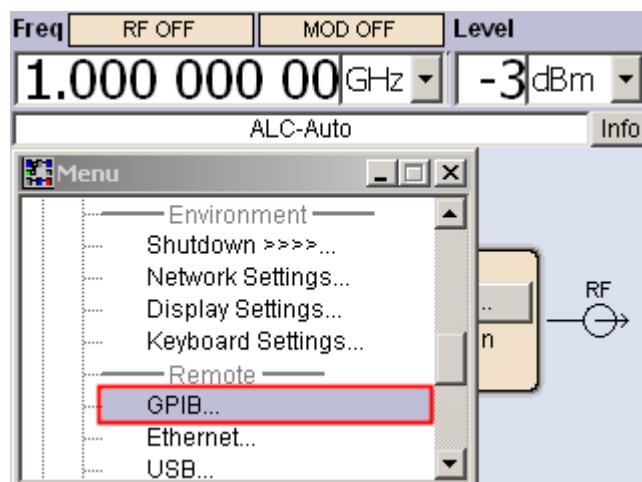
In the Remote Channel Settings dialog, set up the remote emulation specific parameters:
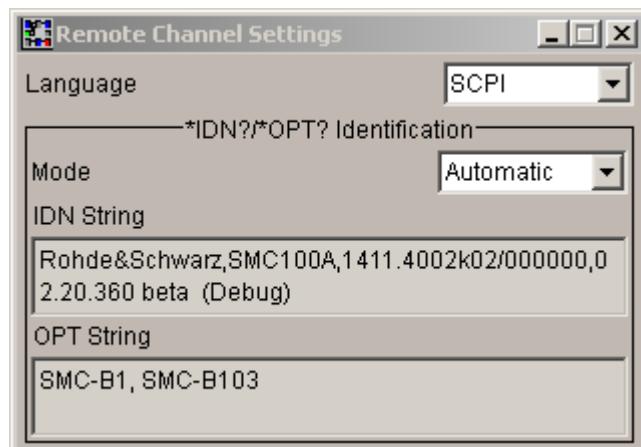
Select the Language and open the drop-down list. Pick an item from the list and confirm the selection:
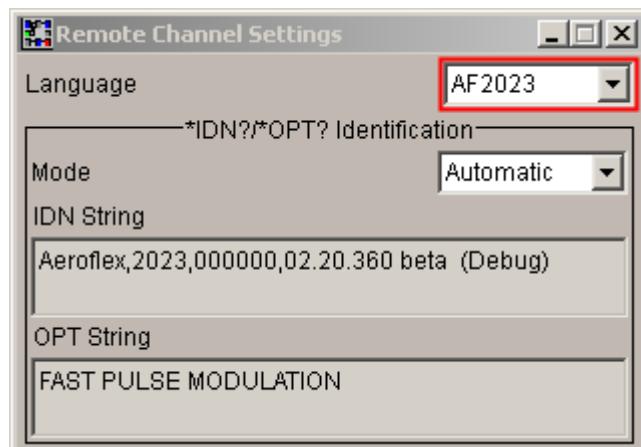
Select the Mode and open the drop-down list. Pick either "Automatic" or "User defined" from the list and confirm the selection. In "Automatic" mode, the response text to *IDN? and *OPT? queries is generated by the signal generator itself. In "User defined" mode, the response text to *IDN? and *OPT? queries must be entered in the corresponding IDN string and/or OPT string text boxes:
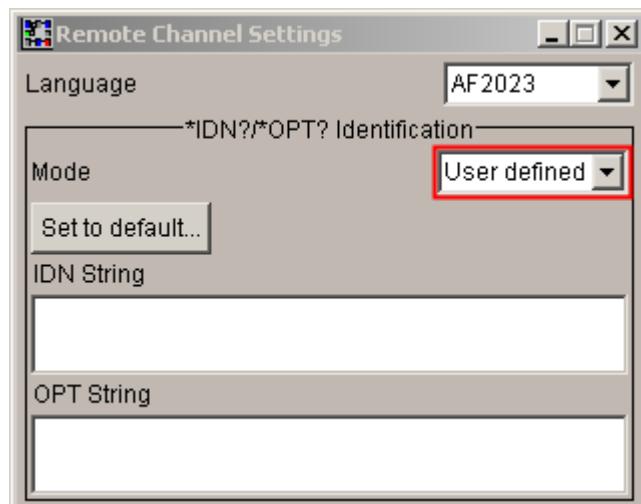
In "User defined" mode, press the "Set to default…" button to preset the corresponding IDN string and/or OPT string text boxes with the default settings from the selected remote emulation:
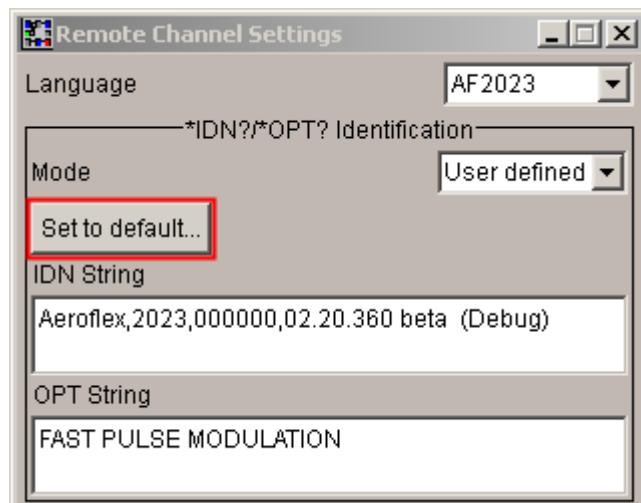
Then select the IDN String text box and edit the response text to *IDN? queries. When finished, confirm the text:
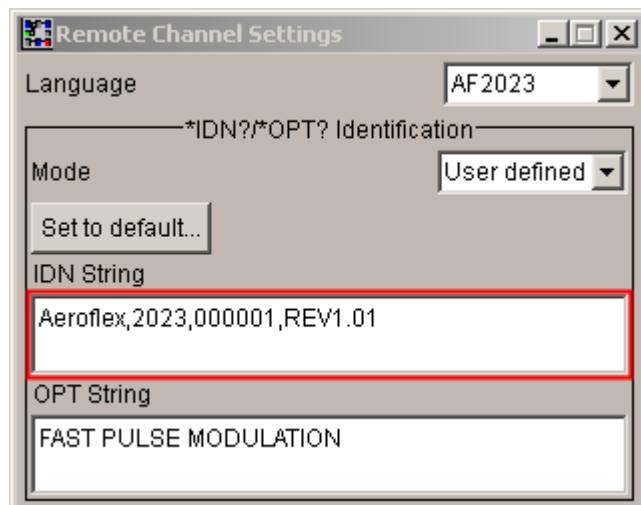
Finally select the OPT String text box and edit the response text to *OPT? queries. When finished, confirm the text:
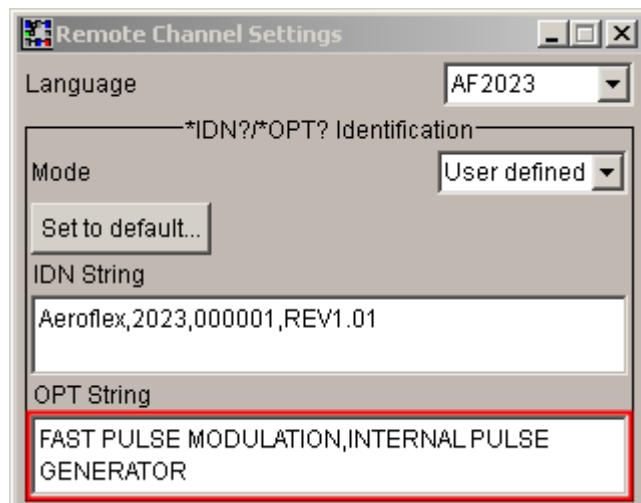
Exit the Remote Channel Settings dialog by pressing the ESC key and verify the selected remote emulation on the SMBV front panel:
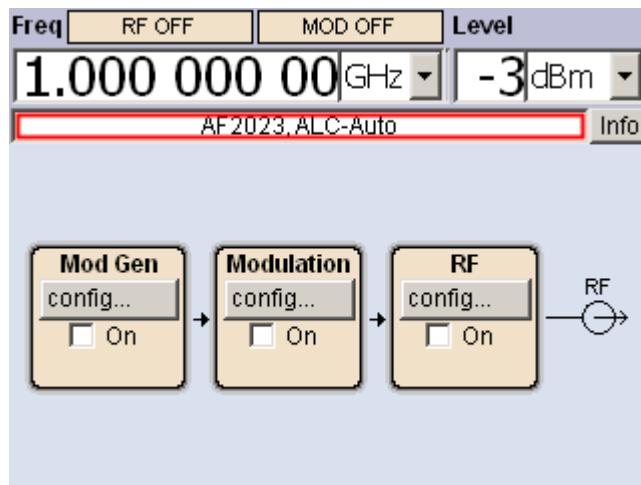
4.2 Remote Operation

When the SMC uses a non-SCPI-compatible language, the remote emulation cannot be changed remotely. The emulation needs to be changed manually.

When the SMC uses an SCPI-compatible language, use the following commands to modify the remote emulation parameters:

| Commands to modify remote emulation relevant settings	
Command	**Comment**
:SYSTem:IDENt <value>	Sets the state of the identification mode.
	If the identification mode is set to USER, the value provided with the command :SYSTem:IDN <value> is returned on an *IDN? query and the value provided with the command :SYSTem:OPT <value> is returned on *OPT? query.
	If the identification mode is set to AUTO, the factory default setting is returned on a *IDN? query or an *OPT? query.
	The value range of character-type parameter <value> is
	• AUTO
	• USER
:SYSTem:IDENt?	Gets the state of the identification mode.
:SYSTem:IDN <value>	Sets the user-defined response to an *IDN? query.
	The string-type parameter <value> allows up to 128 characters. The parameter has to be enclosed in single or double quotes.
:SYSTem:IDN?	Gets the user-defined response to an *IDN? query.
:SYSTem:OPT <value>	Sets the user-defined response to a *OPT? query.
	The string-type parameter <value> allows up to 128 characters. The parameter has to be enclosed in single or double quotes.
:SYSTem:OPT?	Gets the user-defined response to a *OPT? query.

Commands to modify remote emulation relevant settings	
Command	**Comment**
:SYSTem:LANGuage <value>	Activates the remote emulation to be used for further communications. The value range of the string-type parameter <value> is: • "AF2023", "AF2024" • "AF2030", "AF2031", "AF2032", "AF2040", "AF2041", "AF2042" • "E4428", "N5161", "N5181" • "HP8642" • "HP8643", "HP8643", "HP8644", "HP8664", "HP8665" • "HP8647", "HP8648" • "HP8656", "HP8657" • "PA8303" • "SME02", "SME03", "SME06" • "SML01", "SML02", "SML03" • "SMT02", "SMT03", "SMT06" • "EXIT" **Attention:** The remote emulation is changed immediately after parsing this command. Succeeding commands such as *WAI, *OPC or *OPC? are not allowed, since these commands may not be a part of the newly selected command set. **Therefore, this command must be the one and only command in a program message unit.** After sending this command, a delay of two seconds must be applied to the application software before the next command is sent. The parameter value "EXIT" must be used to return to the native SCPI command set of the instrument.
:SYSTem:LANGuage?	Gets the current active remote emulation.

Note:

The upper-case and lower-case notation serves to distinguish between the long and the short form of a command. The instrument itself does not distinguish between upper-case and lower-case notation.

5 Emulating the Aeroflex 2023/2024

The following tables show the current implementation status of each command. Commands not shown in these tables are not supported.

IEEE488.2 functions	
Command syntax	**Status**
*CLS	✔
*ESE value *ESE? **Remark:** Layout of ESE register corresponds to Rohde & Schwarz signal generator.	✋
*ESR? **Remark:** Layout of ESE register corresponds to Rohde & Schwarz signal generator.	✋
*IDN? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	✋
*OPC *OPC?	✔
*OPT? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	✋
*RST	✔
*SRE value *SRE? **Remark:** Layout of ESE register corresponds to Rohde & Schwarz signal generator.	✋
*STB? **Remark:** Layout of ESE register corresponds to Rohde & Schwarz signal generator.	✋
*TRG	✔
*TST?	✔
*WAI	✔

Device-specific functions	
Command syntax	**Status**
AM[1][:DEPTH] value	✔

Device-specific functions	
Command syntax	**Status**
AM[1]:DN	✔
AM[1]:EXTAC	✔
AM[1]:EXTALC	○
AM[1]:EXTDC	✔
AM[1]:INC value	✔
AM[1]:INT	✔
AM[1]:MODF[:VALUE] value	✔
AM[1]:MODF:DN	✔
AM[1]:MODF:INC value	✔
AM[1]:MODF:PHASE value	○
AM[1]:MODF:RETN	✔
AM[1]:MODF:SIN	✔
AM[1]:MODF:SQR	○
AM[1]:MODF:TRI	○
AM[1]:MODF:UP	✔
AM[1]:MODF:XFER	✔
AM[1]:OFF	✔
AM[1]:ON	✔
AM[1]:RETN	✔
AM[1]:UP	✔
AM[1]:XFER	✔
AM[1]?	✔
BLANK OFF	✔
BLANK ON	✔
BLANK?	✔
CCR?	○
CFRQ[:VALUE] value	✔
CFRQ:DN	✔
CFRQ:INC value	✔
CFRQ:MODE	✔
CFRQ:PHASE value	✔
CFRQ:RETN	✔
CFRQ:START	✔

Device-specific functions	
Command syntax	**Status**
CFRQ:STOP	✔
CFRQ:TIME	✔
CFRQ:UP	✔
CFRQ:XFER	✔
CFRQ?	✔
CSE value CSE?	○
CSR?	○
DCFMNL	✔
ERROR? **Remark:** Returns the original error message of the Rohde & Schwarz signal generator.	✋
FM[1][:DEVN] value	✔
FM[1]:DN	✔
FM[1]:EXTAC	✔
FM[1]:EXTALC	○
FM[1]:EXTDC	✔
FM[1]:INC value	✔
FM[1]:INT	✔
FM[1]:MODF[:VALUE] value	✔
FM[1]:MODF:DN	✔
FM[1]:MODF:INC value	✔
FM[1]:MODF:PHASE value	○
FM[1]:MODF:RETN	✔
FM[1]:MODF:SIN	✔
FM[1]:MODF:SQR	○
FM[1]:MODF:TRI	○
FM[1]:MODF:UP	✔
FM[1]:MODF:XFER	✔
FM[1]:OFF	✔
FM[1]:ON	✔
FM[1]:RETN	✔
FM[1]:UP	✔

Device-specific functions	
Command syntax	**Status**
FM[1]:XFER	✓
FM[1]?	✓
FSTD value **Remark:** INT, EXT10DIR are supported.	✋
FSTD?	✓
GPIB value	✓
HCR?	○
HSE value HSE?	○
HSR?	○
KLOCK	✓
KUNLOCK	✓
MODE value **Remark:** Only the following mode combinations are supported: AM1 FM1 PM1 PULSE AM1, FM1 AM1, PM1 PULSE,FM1 PULSE,PM1 PULSE, FM1 PULSE, PM1	✋
MODE?	✓
MOD:OFF	✓
MOD:ON	✓
MOD?	✓
OPER?	✓
OUTPUT:DISABLE	○
OUTPUT:ENABLE	○
PM[1][:DEVN] value	✓
PM[1]:DN	✓
PM[1]:EXTAC	✓

Device-specific functions	
Command syntax	**Status**
PM[1]:EXTALC	◯
PM[1]:EXTDC	✓
PM[1]:INC value	✓
PM[1]:INT	✓
PM[1]:MODF[:VALUE] value	✓
PM[1]:MODF:DN	✓
PM[1]:MODF:INC value	✓
PM[1]:MODF:PHASE value	◯
PM[1]:MODF:RETN	✓
PM[1]:MODF:SIN	✓
PM[1]:MODF:SQR	◯
PM[1]:MODF:TRI	◯
PM[1]:MODF:UP	✓
PM[1]:MODF:XFER	✓
PM[1]:OFF	✓
PM[1]:ON	✓
PM[1]:RETN	✓
PM[1]:UP	✓
PM[1]:XFER	✓
PM[1]?	✓
PULSE:EXT	✓
PULSE:INT	✓
PULSE::MODF[:VALUE] value	✓
PULSE:OFF	✓
PULSE:ON	✓
PULSE?	✓
RFLV[:VALUE] value	✓
RFLV:DN	✓
RFLV:INC value	✓
RFLV:LIMIT:DISABLE	✓
RFLV:LIMIT:ENABLE	✓
RFLV:LIMIT[:VALUE] value	✓
RFLV:LIMIT:SAVE	◯

Device-specific functions	
Command syntax	**Status**
RFLV:OFF	✔
RFLV:OFFS:DISABLE	✔
RFLV:OFFS:ENABLE	✔
RFLV:OFFS:SAVE	◯
RFLV:OFFS:VALUE value	✔
RFLV:ON	✔
RFLV:RETN	✔
RFLV:TYPE value	✔
RFLV:UNITS value	✔
RFLV:UP	✔
RFLV:XFER	✔
RFLV?	✔
RFLV:LIMIT?	✔
RFLV:OFFS?	✔
SCR?	◯
SSE value SSE?	◯
SSR?	◯
SWEEP:CFRQ:INC value	✔
SWEEP:CFRQ:LOGINC value	✔
SWEEP:CFRQ:START value	✔
SWEEP:CFRQ:STOP value	✔
SWEEP:CFRQ:TIME value	✔
SWEEP:CONT **Remark:** Identical to SWEEP:GO.	✋
SWEEP:DN	✔
SWEEP:GO	✔
SWEEP:HALT	✔
SWEEP:MODE value	✔
SWEEP:RESET	✔
SWEEP:TRIG value **Remark:** OFF, START, STEP are supported.	✋

Device-specific functions	
Command syntax	**Status**
SWEEP:TYPE value	✔
SWEEP:UP	✔
SWEEP:XFER	○
SWEEP?	✔
SWEEP:CFRQ?	✔

6 Emulating the Aeroflex 2030-2032, 2040-2042

The following tables show the current implementation status of each command. Commands not shown in these tables are not supported.

IEEE488.2 functions	
Command syntax	**Status**
*CLS	✓
*ESE value *ESE? **Remark:** Layout of ESE register corresponds to Rohde & Schwarz signal generator.	🖐
*ESR? **Remark:** Layout of ESE register corresponds to Rohde & Schwarz signal generator.	🖐
*IDN? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	🖐
*OPC *OPC?	✓
*OPT? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	🖐
*RST	✓
*SRE value *SRE? **Remark:** Layout of ESE register corresponds to Rohde & Schwarz signal generator.	🖐
*STB? **Remark:** Layout of ESE register corresponds to Rohde & Schwarz signal generator.	🖐
*TRG	✓
*TST?	✓
*WAI	✓

Device-specific functions	
Command syntax	**Status**
AM[1] [:DEPTH] value	✓
AM[1]:DN	✓
AM[1]:EXT1AC	✓
AM[1]:EXT1DC	✓
AM[1]:INC value	✓
AM[1]:INTF1	✓
AM[1]:INTF2	✓
AM[1]:INTF3	✓
AM[1]:INTF4	✓
AM[1]:INTF5	✓
AM[1]:INTF6	✓
AM[1]:OFF	✓
AM[1]:ON	✓
AM[1]:RETN	✓
AM[1]:UP	✓
AM[1]:XFER	✓
AM[1]?	✓
BLANK value **Remark:** 0, 1, 4 are supported.	✋
BACKL:ON	✓
BACKL:OFF	✓
CFRQ[:VALUE] value	✓
CFRQ:DN	✓
CFRQ:INC value	✓
CFRQ:PHASE value	✓
CFRQ:RETN	✓
CFRQ:UP	✓
CFRQ:XFER	✓
CFRQ?	✓
CCR?	○
CSE value CSE?	○

Device-specific functions	
Command syntax	**Status**
CSR?	◯
DATE?	✓
DCFMNL	✓
ELAPSED?	✓
ERROR? **Remark:** Returns the original error message of the Rohde & Schwarz signal generator.	✋
FM[1][:DEVN] value	✓
FM[1]:DN	✓
FM[1]:EXT1AC	✓
FM[1]:EXT1DC	✓
FM[1]:INC value	✓
FM[1]:INTF1	✓
FM[1]:INTF2	✓
FM[1]:INTF3	✓
FM[1]:INTF4	✓
FM[1]:INTF5	✓
FM[1]:INTF6	✓
FM[1]:OFF	✓
FM[1]:ON	✓
FM[1]:RETN	✓
FM[1]:UP	✓
FM[1]:XFER	✓
FM[1]?	✓
FSTD value **Remark:** INT10 , EXT5, EXT10 are supported.	✋
FSTD?	✓
HCR?	◯
HSE value HSE?	◯
HSR?	◯

Device-specific functions	
Command syntax	**Status**
IMODE value **Remark:** NORMAL, SWEEPER are supported.	🖐
INTF1|2|3|4|5|6[:FREQ] value	✓
INTF1|2|3|4|5|6:CTC1	○
INTF1|2|3|4|5|6:CTC2	○
INTF1|2|3|4|5|6:DN	✓
INTF1|2|3|4|5|6:INC value	✓
INTF1|2|3|4|5|6:PHASE value	○
INTF1|2|3|4|5|6:SIN	✓
INTF1|2|3|4|5|6:SQU	○
INTF1|2|3|4|5|6:RETN	○
INTF1|2|3|4|5|6:TEMP	○
INTF1|2|3|4|5|6:TRI	○
INTF1|2|3|4|5|6:UP	✓
INTF1|2|3|4|5|6:USER	○
INTF1|2|3|4|5|6:XFER	✓
INTF1|2|3|4|5|6?	✓
KLOCK	✓
KUNLOCK	✓
LF:ON	✓
LF:OFF	✓
LF?	✓
LFGF[:VALUE] value	✓
LFGF:DN	✓
LFGF:INC value	✓
LFGF:RETN	✓
LFGF:SIN	✓
LFGF:SQU	○
LFGF:TRI	○
LFGF:UP	✓
LFGF:XFER	✓
LFGF?	✓

Device-specific functions	
Command syntax	**Status**
LFGL[:VALUE] value	✓
LFGL:DN	✓
LFGL:INC value	✓
LFGL:RETN	✓
LFGL:UNITS value **Remark:** V, MV, UV are not supported.	🖑
LFGL:UP	✓
LFGL:XFER	✓
LFGL?	✓
MODE value **Remark:** Only the following mode combinations are supported: AM1 FM1 PM1 PULSE AM1, FM1 AM1, PM1 PULSE,FM1 PULSE,PM1	🖑
MODE?	✓
MOD:OFF	✓
MOD:ON **Remark:** Re-enables previously enabled analogous modulations.	🖑
MOD?	✓
OPER?	✓
PGEN:DELAY value	○
PGEN:D_DELAY value	✓
PGEN:RATE value	✓
PGEN:SOURCE value	✓
PGEN:STATE value	✓
PGEN:TRIGGER value	✓
PGEN:WIDTH value	✓
PM[1][:DEVN] value	✓

Device-specific functions	
Command syntax	**Status**
PM[1]:DN	✔
PM[1]:EXT1AC	✔
PM[1]:EXT1DC	✔
PM[1]:INC value	✔
PM[1]:INTF1	✔
PM[1]:INTF2	✔
PM[1]:INTF3	✔
PM[1]:INTF4	✔
PM[1]:INTF5	✔
PM[1]:INTF6	✔
PM[1]:OFF	✔
PM[1]:ON	✔
PM[1]:RETN	✔
PM[1]:UP	✔
PM[1]:XFER	✔
PM[1]?	✔
PULSE:CAL:ENABLE	○
PULSE:CAL:DISABLE	○
PULSE:OFF	✔
PULSE:ON	✔
PULSE?	✔
PULSE:CAL?	○
RFLV[:VALUE] value	✔
RFLV:DN	✔
RFLV:HYST:DISABLE	○
RFLV:HYST:ENABLE	○
RFLV:INC value	✔
RFLV:LIMIT[:VALUE] value	✔
RFLV:LIMIT:DISABLE	✔
RFLV:LIMIT:ENABLE	✔
RFLV:LIMIT:SAVE	○
RFLV:OFF	✔
RFLV:OFFS:DISABLE	✔

Device-specific functions	
Command syntax	**Status**
RFLV:OFFS:ENABLE	✔
RFLV:OFFS:OFF	✔
RFLV:OFFS:ON	✔
RFLV:OFFS:SAVE	○
RFLV:OFFS:VALUE value	✔
RFLV:ON	✔
RFLV:RETN	✔
RFLV:TYPE value	✔
RFLV:UNITS value	✔
RFLV:UP	✔
RFLV:XFER	✔
RFLV?	✔
RFLV:HYST?	○
RFLV:LIMIT?	✔
RFLV:OFFS?	✔
SCR?	○
SSE value SSE?	○
SSR?	○
SWEEP:CALC	○
SWEEP:CFRQ:STARTvalue	✔
SWEEP:CFRQ:STEP value	✔
SWEEP:CFRQ:STOP value	✔
SWEEP:CFRQ:TIME value	✔
SWEEP:CONT **Remark:** Identical to SWEEP:GO.	✋
SWEEP:GO	✔
SWEEP:HALT	✔
SWEEP:INTF:STARTvalue	✔
SWEEP:INTF:STEP value	✔
SWEEP:INTF:STOP value	✔
SWEEP:INTF:TIME value	✔

Device-specific functions	
Command syntax	**Status**
SWEEP:LFGF:STARTvalue	✔
SWEEP:LFGF:STEP value	✔
SWEEP:LFGF:STOP value	✔
SWEEP:LFGF:TIME value	✔
SWEEP:MODE value	✔
SWEEP:RESET	✔
SWEEP:RFLV:STARTvalue	✔
SWEEP:RFLV:STEP value	✔
SWEEP:RFLV:STOP value	✔
SWEEP:RFLV:TIME value	✔
SWEEP:TYPE value **Remark:** CFRQ, RFLV, LFGF, INTF1, INTF2, INTF3, INTF4, INTF5, INTF6 are supported.	🖐
SWEEP?	✔
SWEEP:CFRQ?	✔
SWEEP:INTF?	✔
SWEEP:LFGF?	✔
SWEEP:RFLV?	✔
SWEEP?	✔
TIME?	✔

7 Emulating the Agilent E4428, N5161, N5181

The file system implemented in the E4428 / N5161 / N5181 is not compatible with the file system implemented in the SMC. The SMC does not support RAM-based file storage; all file contents are stored persistently.

Furthermore, the E4428 / N5161 / N5181 directory tree is not compatible with the SMC directory tree. The following tables show the mapping of the E4428 / N5161 / N5181 directory tree to the SMC directory tree:

Directory tree mapping	
E4428	**SMC**
/user/list/	/var/user/Agilent/Esg/user/list/
/user/userflat/	/var/user/Agilent/Esg/user/userflat/

Directory tree mapping	
N5161 / N5181	**SMC**
/user/list/	/var/user/Agilent/Mxg/user/list/
/user/userflat/	/var/user/Agilent/Mxg/user/userflat/

When E4428 / N5161 / N5181 remote application is applied, the corresponding SMC directory tree is created automatically.

The file format of the E4428 / N5161 / N5181 is different than the file format used in the SMC. Therefore,

- E4428 / N5161 / N5181 formatted files must not be transferred via FTP or USB memory to the SMC.
- E4428 / N5161 / N5181 formatted files must be transferred via remote interface, since the remote emulation performs the required conversion to the SMC file format.

The following tables show the current implementation status of each command. Commands not shown in these tables are not supported.

IEEE488.2 functions	
Command syntax	**Status**
*CLS	✓
*ESE value *ESE? **Remark:** Layout of ESE register corresponds to Rohde & Schwarz signal generator.	🖐

IEEE488.2 functions	
Command syntax	**Status**
*ESR? **Remark:** Layout of ESE register corresponds to Rohde & Schwarz signal generator.	🖐
*IDN? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	🖐
*OPC *OPC?	✓
*OPT? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	🖐
*RST	✓
*SRE value *SRE? **Remark:** Layout of ESE register corresponds to Rohde & Schwarz signal generator.	🖐
*STB? **Remark:** Layout of ESE register corresponds to Rohde & Schwarz signal generator.	🖐
*TRG	✓
*TST?	✓
*WAI	✓

Device-specific functions	
Command syntax	**Status**
ABORt	✓
CALibration:DCFM	✓
CALibration:IQ:FULL	✓
DISPlay:ANNotation:AMPLitude[:STATe] value DISPlay:ANNotation:AMPLitude[:STATe]?	✓
DISPlay:ANNotation:FREQuency[:STATe] value DISPlay:ANNotation:FREQuency[:STATe]?	✓
DISPlay:REMote value DISPlay:REMote?	✓
INITiate:CONTinuous[:ALL] value INITiate:CONTinuous[:ALL]?	◯

Device-specific functions	
Command syntax	**Status**
INITiate:IMMediate[:ALL]	○
MEMory:CATalog[:ALL]?	✓
MEMory:CATalog:LIST?	✓
MEMory:CATalog:ULFT?	✓
MEMory:COPY[:NAME] value, value	✓
MEMory:DATA value,value MEMory:DATA? value	✓
MEMory:DATA:UNPRotected value, value MEMory:DATA:UNPRotected? Value **Remark:** Implementation identical to MEMory:DATA[?].	🖐
MEMory:DELete:ALL	✓
MEMory:DELete:LIST	✓
MEMory:DELete[:NAME] value	✓
MEMory:DELete:ULFT	✓
MEMory:FREE[:ALL]	✓
MEMory:LOAD:LIST value	✓
MEMory:MOVE	✓
MEMory:STORE:LIST value	✓
MMEMory:CATalog[:ALL]? value	✓
MMEMory:COPY value, value	✓
MMEMory:DATA value, value MMEMory:DATA? value	✓
MMEMory:DELete[:NAME] value	✓
MMEMory:LOAD:LIST value	✓
MMEMory:LOAD:MOVE value, value	✓
MMEMory:STORe:LIST value	✓
OUTPut:MODulation[:STATe] value OUTPut:MODulation[:STATe]?	✓
OUTPut[:STATe] value OUTPut[:STATe]?	✓
[SOURce:]AM[1] [:DEPth][:LINear] value [SOURce:]AM[1] [:DEPth][:LINear]?	✓
[SOURce:]AM[1] [:DEPth]:STEP[:INCRement] value [SOURce:]AM[1] [:DEPth]:STEP[:INCRement]?	✓

Device-specific functions	
Command syntax	**Status**
[SOURce:]AM[1]:EXTernal[1]:COUPling value [SOURce:]AM[1]:EXTernal[1]:COUPling?	✓
[SOURce:]AM[1]:INTernal[1]:FREQuency value [SOURce:]AM[1]:INTernal[1]:FREQuency?	✓
[SOURce:]AM[1]:INTernal[1]:FREQuency:STEP[:INCRement] value [SOURce:]AM[1]:INTernal[1]:FREQuency:STEP[:INCRement]?	✓
[SOURce:]AM[1]:INTernal[1]:FUNCtion:SHAPe value [SOURce:]AM[1]:INTernal[1]:FUNCtion:SHAPe? **Remark:** SINE is supported.	🖐
[SOURce:]AM[1]:INTernal[1]:SWEep:TIME value [SOURce:]AM[1]:INTernal[1]:SWEep:TIME? **Remark:** Dwell time is assumed.	🖐
[SOURce:]AM[1]:INTernal[1]:SWEep:TRIGger value [SOURce:]AM[1]:INTernal[1]:SWEep:TRIGger? **Remark:** BUS, EXTernal, IMMediate are supported.	🖐
[SOURce:]AM[1]:SOURce value [SOURce:]AM[1]:SOURce? **Remark:** EXTernal[1], INTernal[1] are supported.	🖐
[SOURce:]AM[1]:STATe value [SOURce:]AM[1]:STATe?	✓
[SOURce:]CORRection:FLATness:LOAD value	✓
[SOURce:]CORRection:FLATness:PAIR value, value {value, value} [SOURce:]CORRection:FLATness:PAIR?	✓
[SOURce:]CORRection:FLATness:POINts?	✓
[SOURce:]CORRection:FLATness:PRESet	✓
[SOURce:]CORRection:FLATness:STORe value	✓
[SOURce:]CORRection[:STATe] value [SOURce:]CORRection[:STATe]?	✓
[SOURce:]FM[1][:DEViation] value [SOURce:]FM[1][:DEViation]? **Remark:** DOWN, MINImum, Numeric, UP are supported.	🖐
[SOURce:]FM[1][:DEViation]:STEP[:INCRement] value [SOURce:]FM[1][:DEViation]:STEP[:INCRement]?	✓

Device-specific functions	
Command syntax	**Status**
[SOURce:]FM[1]|2:EXTernal[12:COUPling value [SOURce:]FM[1]|2:EXTernal[1]:COUPling?	✔
[SOURce:]FM[1]:INTernal[1]:FREQuency value [SOURce:]FM[1]:INTernal[1]:FREQuency?	✔
[SOURce:]FM[1]:INTernal[1]:FREQuency:STEP[:INCRement] value [SOURce:]FM[1]:INTernal[1]:FREQuency:STEP[:INCRement]?	✔
[SOURce:]FM[1]:INTernal[1]:FUNCtion:SHAPe value [SOURce:]FM[1]:INTernal[1]:FUNCtion:SHAPe? **Remark:** SINE is supported.	🖐
[SOURce:]FM[1]:INTernal[1]:SWEep:TIME value [SOURce:]FM[1]:INTernal[1]:SWEep:TIME? **Remark:** Dwell time is assumed.	🖐
[SOURce:]FM[1]:INTernal[1]:SWEep:TRIGger value [SOURce:]FM[1]:INTernal[1]:SWEep:TRIGger? **Remark:** BUS, EXTernal, IMMediate are supported.	🖐
[SOURce:]FM[1]:SOURce value [SOURce:]FM[1]:SOURce? **Remark:** EXTernal[1], INTernal[1] are supported.	🖐
[SOURce:]FM[1]:STATe value [SOURce:]FM[1]:STATe?	✔
[SOURce:]FREQuency[:CW] value [SOURce:]FREQuency[:CW]?	✔
[SOURce:]FREQuency[:CW]:STEP[:INCRement] value [SOURce:]FREQuency[:CW]:STEP[:INCRement]?	✔
[SOURce:]FREQuency:FIXed value [SOURce:]FREQuency:FIXed?	✔
[SOURce:]FREQuency:MODE [SOURce:]FREQuency:MODE?	✔
[SOURce:]FREQuency:OFFSet value [SOURce:]FREQuency:OFFSet?	✔
[SOURce:]FREQuency:OFFSet:STATe value [SOURce:]FREQuency:OFFSet:STATe?	✔
[SOURce:]FREQuency:REFerence value [SOURce:]FREQuency:REFerence?	✔

Device-specific functions	
Command syntax	**Status**
[SOURce:]FREQuency:REFerence:STATe value [SOURce:]FREQuency:REFerence:STATe?	✔
[SOURce:]FREQuency:STARt value [SOURce:]FREQuency:STARt?	✔
[SOURce:]FREQuency:STOP value [SOURce:]FREQuency:STOP?	✔
[SOURce:]LFOutput:AMPLitude value [SOURce:]LFOutput:AMPLitude?	✔
[SOURce:]LFOutput:FUNCtion[1]:FREQuency value [SOURce:]LFOutput:FUNCtion[1]:FREQuency?	✔
[SOURce:]LFOutput:FUNCtion[1]:FREQuency:STEP[:INCRement] value [SOURce:]LFOutput:FUNCtion[1]:FREQuency:STEP[:INCRement]?	✔
[SOURce:]LFOutput:FUNCtion[1]:PERiod value [SOURce:]LFOutput:FUNCtion[1]:PERiod?	✔
[SOURce:]LFOutput:FUNCtion[1]:PERiod:STEP[:INCRement] value [SOURce:]LFOutput:FUNCtion[1]:PERiod:STEP[:INCRement]?	✔
[SOURce:]LFOutput:FUNCtion[1]:PWIDth value [SOURce:]LFOutput:FUNCtion[1]:PWIDth?	✔
[SOURce:]LFOutput:FUNCtion[1]:PWIDth:STEP[:INCRement] value [SOURce:]LFOutput:FUNCtion[1]:PWIDth:STEP[:INCRement]?	✔
[SOURce:]LFOutput:FUNCtion[1]:SHAPe value [SOURce:]LFOutput:FUNCtion[1]:SHAPe? **Remark:** SINE is supported.	✋
[SOURce:]LFOutput:FUNCtion[1]:SWEep:TRIGger value [SOURce:]LFOutput:FUNCtion[1]:SWEep:TRIGger? **Remark:** BUS, EXTernal, IMMediate are supported.	✋
[SOURce:]LFOutput:FUNCtion[1]:SOURce value [SOURce:]LFOutput:FUNCtion[1]:SOURce? **Remark:** INTernal[1] is supported.	✋
[SOURce:]LFOutput:FUNCtion[1]:STATe value [SOURce:]LFOutput:FUNCtion[1]:STATe?	✔
[SOURce:]LIST:DIRection?	○

Device-specific functions	
Command syntax	**Status**
[SOURce:]LIST:DWELl value {,value} [SOURce:]LIST:DWELl? **Remark:** First parameter value is evaluated, rest is ignored.	✋
[SOURce:]LIST:DWELl:POINts?	✓
[SOURce:]LIST:DWELl:TYPE value [SOURce:]LIST:DWELl:TYPE?	✓
[SOURce:]LIST:FREQuency value {,value} [SOURce:]LIST:FREQuency?	✓
[SOURce:]LIST:FREQuency:POINts?	✓
[SOURce:]LIST:MODE value [SOURce:]LIST:MODE?	✓
[SOURce:]LIST:POWer value {,value} [SOURce:]LIST:POWer?	✓
[SOURce:]LIST:POWer:POINts?	✓
[SOURce:]LIST:TYPE value [SOURce:]LIST:TYPE?	✓
[SOURce:]LIST:TYPE:LIST:INITialize:FSTep	✓
[SOURce:]LIST:TYPE:LIST:INITialize:PREset	✓
[SOURce:]PHASe[:ADJust] value [SOURce:]PHASe[:ADJust]?	✓
[SOURce:]PHASe:REFerence	✓
[SOURce:]PM[1][:DEViation] value [SOURce:]PM[1][:DEViation]? **Remark:** DOWN, MINImum, Numeric, UP are supported.	✋
[SOURce:]PM[1][:DEViation]:STEP[:INCRement] value [SOURce:]PM[1][:DEViation]:STEP[:INCRement]?	✓
[SOURce:]PM[1]:EXTernal[1]:COUPling value [SOURce:]PM[1]:EXTernal[1]:COUPling?	✓
[SOURce:]PM[1]:INTernal[1]:FREQuency value [SOURce:]PM[1]:INTernal[1]:FREQuency?	✓
[SOURce:]PM[1]:INTernal[1]:FREQuency:STEP[:INCRement] value [SOURce:]PM[1]:INTernal[1]:FREQuency:STEP[:INCRement]?	✓

Device-specific functions	
Command syntax	**Status**
[SOURce:]PM[1]:INTernal[1]:FUNCtion:SHAPe value [SOURce:]PM[1]:INTernal[1]:FUNCtion:SHAPe? **Remark:** SINE is supported.	✋
[SOURce:]PM[1]:INTernal[1]:SWEep:TIME value [SOURce:]PM[1]:INTernal[1]:SWEep:TIME? **Remark:** Dwell time is assumed.	✋
[SOURce:]PM[1]:INTernal[1]:SWEep:TRIGger value [SOURce:]PM[1]:INTernal[1]:SWEep:TRIGger? **Remark:** BUS, EXTernal, IMMediate are supported.	✋
[SOURce:]PM[1]:SOURce value [SOURce:]PM[1]:SOURce? **Remark:** EXTernal[1], INTernal[1] are supported.	✋
[SOURce:]FM[1]:STATe value [SOURce:]FM[1]:STATe?	✓
[SOURce:]POWer:ALC[:STATe] value [SOURce:]POWer:ALC[:STATe]?	✓
[SOURce:]POWer:ATTenuation value [SOURce:]POWer:ATTenuation?	✓
[SOURce:]POWer:ATTenuation:AUTO value [SOURce:]POWer:ATTenuation:AUTO?	✓
[SOURce:]POWer[:LEVel][:IMMediate][:AMPLitude] value [SOURce:]POWer[:LEVel][:IMMediate][:AMPLitude]?	✓
[SOURce:]POWer[:LEVel][:IMMediate][:AMPLitude]:STEP[:INCRement] value [SOURce:]POWer[:LEVel][:IMMediate][:AMPLitude]:STEP[:INCRement]?	✓
[SOURce:]POWer[:LEVel][:IMMediate]:OFFSet value [SOURce:]POWer[:LEVel][:IMMediate]:OFFSet?	✓
[SOURce:]POWer:MODE value [SOURce:]POWer:MODE?	✓
[SOURce:]POWer:REFerence value [SOURce:]POWer:REFerence?	✓
[SOURce:]POWer:REFerence:STATe value [SOURce:]POWer:REFerence:STATe?	✓
[SOURce:]POWer:STARt value [SOURce:]POWer:STARt?	✓

Device-specific functions	
Command syntax	**Status**
[SOURce:]POWer:STOP value [SOURce:]POWer:STOP?	✓
[SOURce:]PULM:EXTernal:POLarity value [SOURce:]PULM:EXTernal:POLarity?	✓
[SOURce:]PULM:INTernal[1]:FREQuency value [SOURce:]PULM:INTernal[1]:FREQuency?	✓
[SOURce:]PULM:INTernal[1]:FREQuency:STEP[:INCRement] value [SOURce:]PULM:INTernal[1]:FREQuency:STEP[:INCRement]?	✓
[SOURce:]PULM:INTernal[1]:FUNCtion:SHAPe value [SOURce:]PULM:INTernal[1]:FUNCtion:SHAPe? **Remark:** SQUare is supported.	🖐
[SOURce:]PULM:INTernal[1]:PERiod value [SOURce:]PULM:INTernal[1]:PERiod?	✓
[SOURce:]PULM:INTernal[1]:PERiod:STEP[:INCRement] value [SOURce:]PULM:INTernal[1]:PERiod:STEP[:INCRement]?	✓
[SOURce:]PULM:INTernal[1]:PWIDth value [SOURce:]PULM:INTernal[1]:PWIDth?	✓
[SOURce:]PULM:INTernal[1]:PWIDth:STEP[:INCRement] value [SOURce:]PULM:INTernal[1]:PWIDth:STEP[:INCRement]?	✓
[SOURce:]PULM:INTernal[1]:SOURce value [SOURce:]PULM:INTernal[1]:SOURce? **Remark:** EXTernal[1], INTernal are supported.	🖐
[SOURce:]PULM:INTernal[1]:STATe value [SOURce:]PULM:INTernal[1]:STATe?	✓
[SOURce:]:ROSCillator:SOURce value [SOURce:]:ROSCillator:SOURce?	✓
[SOURce:]SWEep:DWELl value [SOURce:]SWEep:DWELl?	✓
[SOURce:]SWEep:POINts value [SOURce:]SWEep:POINts?	✓
STATus:OPERation:CONDition?	✓
STATus:OPERation:ENABle value STATus:OPERation:ENABle?	✓
STATus:OPERation[:EVENt]?	✓

Device-specific functions	
Command syntax	**Status**
STATus:OPERation:NTRansition value STATus:OPERation: NTRansition?	✓
STATus:OPERation:PTRansition value STATus:OPERation: PTRansition?	✓
STATus:PRESet	✓
STATus:QUEStionable:CONDition?	✓
STATus:QUEStionable:ENABle value STATus:QUEStionable:ENABle?	✓
STATus:QUEStionable[:EVENt]?	✓
STATus:QUEStionable:NTRansition value STATus:QUEStionable: NTRansition?	✓
STATus:QUEStionable:PTRansition value STATus:QUEStionable: PTRansition?	✓
SYSTem:CAPability?	○
SYSTem:COMMunicate:GPIB:ADDRess value SYSTem:COMMunicate:GPIB:ADDRess?	✓
SYSTem:COMMunicate:GTLocal	✓
SYSTem:COMMunicate:LAN:CONFig value SYSTem:COMMunicate:LAN:CONFig?	✓
SYSTem:COMMunicate:LAN:GATeway value SYSTem:COMMunicate:LAN:GATeway?	✓
SYSTem:COMMunicate:LAN:HOSTname value SYSTem:COMMunicate:LAN:HOSTname?	✓
SYSTem:COMMunicate:LAN:IP value SYSTem:COMMunicate:LAN:IP?	✓
SYSTem:COMMunicate:LAN:SUBNet value SYSTem:COMMunicate:LAN:SUBNet?	✓
SYSTem:COMMunicate:SERial:BAUD value SYSTem:COMMunicate:SERial:BAUD?	✓
SYSTem:DATE value SYSTem:DATE?	✓
SYSTem:ERRor:CODE[:NEXT]? **Remark:** Returns the original error code of the Rohde & Schwarz signal generator.	✋
SYSTem:ERRor[:NEXT]? **Remark:** Returns the original error message of the Rohde & Schwarz signal generator.	✋

Device-specific functions	
Command syntax	**Status**
SYSTem:IDN value SYSTem:IDN?	✔
SYSTem:OPT value SYSTem:OPT?	✔
SYSTem:PDOWn	✔
SYSTem:PRESet	✔
SYSTem:PRESet:ALL	✔
SYSTem:SECurity:DISPlay value SYSTem:SECurity:DISPlay?	✔
SYSTem:TIME value SYSTem:TIME?	✔
SYSTem:VERSion?	✔
TRIGger[:SEQuence][:IMMediate]	✔
TRIGger[:SEQuence]:SLOPe value TRIGger[:SEQuence]:SLOPe?	✔
TRIGger[:SEQuence]:SOURce value TRIGger[:SEQuence]:SOURce? **Remark:** BUS, EXTernal, IMMediate are supported.	✋
UNIT:POWer value UNIT:POWer? **Remark:** DB, DBM, DBV, DBMV, DBUV, V, MV, UV are supported.	✋

8 Emulating the Hewlett-Packard 8642

The following tables show the current implementation status of each command. Commands not shown in these tables are not supported.

Device-specific functions	
Command syntax	**Status**
AA value **Remark:** Units DB, DM, DU, MV, VL are supported.	✋
AB value **Remark:** Units DB, DM, DU, MV, VL are supported.	✋
AM value	✔
AP value **Remark:** Units DB, DM, DU, MV, VL are supported.	✋
BA	✔
BD	✔
DN	✔
EMOF	✔
EMON	✔
FA value	✔
FB value	✔
FM value	✔
FR value	✔
IP	✔
IS value **Remark:** HZ, GZ, KZ, MZ, DB are supported.	✋
MF value	✔
ML value	✔
NT	✔
OF	✔
ON	✔
PM value	✔
PL	✔

Device-specific functions	
Command syntax	**Status**
RA value	✔
RF value	✔
XA	✔
XD	✔
UP	✔

9 Emulating the Hewlett-Packard 8643-8645, 8664/8665

The following tables show the current implementation status of each command. Commands not shown in these tables are not supported.

IEEE488.2 functions	
Command syntax	**Status**
*CAL?	✔
*CLS	✔
*ESE value *ESE?	✔
*ESR?	✔
*IDN? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	✋
*OPC *OPC?	✔
*OPT? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	✋
*RCL value	✔
*RST	✔
*SAV value	✔
*SRE value *SRE?	✔
*STB?	✔
*TST?	✔
*WAI	✔

Device-specific functions	
Command syntax	**Status**
[SOURce:]AM:COUPling value [SOURce:]AM:COUPling? **Remark:** AC, DC are supported.	✋

Device-specific functions	
Command syntax	**Status**
[SOURce:]AM[:DEPTh] value [SOURce:]AM[:DEPTh]?	✔
[SOURce:]AM[:DEPTh]:STEP[:INCRement] value [SOURce:]AM[:DEPTh]:STEP[:INCRement]?	✔
[SOURce:]AM:FREQuency value [SOURce:]AM:FREQuency?	✔
[SOURce:]AM:FREQuency:STEP[:INCRement] value [SOURce:]AM:FREQuency:STEP[:INCRement]?	✔
[SOURce:]AM:SOURce value [SOURce:]AM:SOURce?	✔
[SOURce:]AM:STATe value [SOURce:]AM:STATe?	✔
AMPLitude|POWer[:OUT]:ATTenuation value AMPLitude|POWer[:OUT]:ATTenuation?	✔
AMPLitude|POWer[:OUT]:ATTenuation:AUTO value AMPLitude|POWer[:OUT]:ATTenuation:AUTO?	✔
AMPLitude|POWer[:OUT]:GAIN value AMPLitude|POWer[:OUT]:GAIN?	✔
AMPLitude|POWer[:OUT][:LEVel] value AMPLitude|POWer[:OUT][:LEVel]?	✔
AMPLitude|POWer[:OUT][:LEVel]:STEP[:INCRement] value AMPLitude|POWer[:OUT][:LEVel]:STEP[:INCRement]?	✔
AMPLitude|POWer[:OUT][:LEVel]:STEP:UNIT value AMPLitude|POWer[:OUT][:LEVel]:STEP:UNIT?	✔
AMPLitude|POWer[:OUT]:STATe value AMPLitude|POWer[:OUT]:STATe?	✔
AMPLitude|POWer[:OUT]:ULIMit value AMPLitude|POWer[:OUT]:ULIMit?	✔
AMPLitude|POWer[:OUT]:UNIT value AMPLitude|POWer[:OUT]:UNIT?	✔
AMPLitude|POWer:SOURce:ATTenuation value AMPLitude|POWer:SOURce:ATTenuation?	✔
AMPLitude|POWer:SOURce:ATTenuation:AUTO value AMPLitude|POWer:SOURce:ATTenuation:AUTO?	✔
AMPLitude|POWer:SOURce:GAIN value AMPLitude|POWer:SOURce:GAIN?	✔

Device-specific functions	
Command syntax	**Status**
AMPLitude|POWer:SOURce[:LEVel] value AMPLitude|POWer:SOURce[:LEVel]?	✓
AMPLitude|POWer:SOURce[:LEVel]:STEP[:INCRement] value AMPLitude|POWer:SOURce[:LEVel]:STEP[:INCRement]?	✓
AMPLitude|POWer:SOURce[:LEVel]:STEP:UNIT value AMPLitude|POWer:SOURce[:LEVel]:STEP:UNIT?	✓
AMPLitude|POWer:SOURce:STATe value AMPLitude|POWer:SOURce:STATe?	✓
AMPLitude|POWer:SOURce:UNIT value AMPLitude|POWer:SOURce:UNIT?	✓
CALibration:ALL?	✓
DISPlay:ANNotation[:ALL] value DISPlay:ANNotation[:ALL]?	✓
DISPlay:ANNotation:AMPLitude value DISPlay:ANNotation:AMPLitude?	✓
DISPlay:ANNotation:FREQuency value DISPlay:ANNotation:FREQuency?	✓
DISPlay:STATe value DISPlay:STATe?	✓
[SOURce:]FM:COUPling value [SOURce:]FM:COUPling? **Remark:** AC, DC are supported.	✋
[SOURce:]FM[:DEViation] value [SOURce:]FM[:DEViation]? **Remark:** Numeric, UP, DOWN, MINIMUM are supported.	✋
[SOURce:]FM[:DEViation]:STEP[:INCRement] value [SOURce:]FM[:DEViation]:STEP[:INCRement]?	✓
[SOURce:]FM:FREQuency value [SOURce:]FM:FREQuency?	✓
[SOURce:]FM:FREQuency:STEP[:INCRement] value [SOURce:]FM:FREQuency:STEP[:INCRement]?	✓
[SOURce:]FM:SOURce value [SOURce:]FM:SOURce?	✓
[SOURce:]FM:STATe value [SOURce:]FM:STATe?	✓

Device-specific functions	
Command syntax	**Status**
[:SOURce:]FREQuency:CENTer value [:SOURce:]FREQuency:CENTer?	✔
[SOURce:]FREQuency:CENTer:STEP[:INCRement] value [SOURce:]FREQuency:CENTer:STEP[:INCRement]?	✔
[:SOURce:]FREQuency[:CW] value [:SOURce:]FREQuency[:CW]?	✔
[SOURce:]FREQuency[:CW]:STEP[:INCRement] value [SOURce:]FREQuency[:CW]:STEP[:INCRement]?	✔
[:SOURce:]FREQuency:INSTantaneous?	✔
[:SOURce:]FREQuency:MANual value [:SOURce:]FREQuency:MANual?	✔
[:SOURce:]FREQuency:MODE value [:SOURce:]FREQuency:MODE? **Remark:** Handling of INITialize and TRIGger is different.	🖐
[:SOURce:]FREQuency:OFFSet value [:SOURce:]FREQuency:OFFSet?	✔
[:SOURce:]FREQuency:SPAN value [:SOURce:]FREQuency:SPAN?	✔
[SOURce:]FREQuency:SPAN:STEP[:INCRement] value [SOURce:]FREQuency:SPAN:STEP[:INCRement]?	✔
[:SOURce:]FREQuency:STARt value [:SOURce:]FREQuency:STARt?	✔
[SOURce:]FREQuency:STARt:STEP[:INCRement] value [SOURce:]FREQuency:STARt:STEP[:INCRement]?	✔
[:SOURce:]FREQuency:STOP value [:SOURce:]FREQuency:STOP?	✔
[SOURce:]FREQuency:STOP:STEP[:INCRement] value [SOURce:]FREQuency:STOP:STEP[:INCRement]?	✔
INITialize:ABORt **Remark:** Handling of INITialize and TRIGger is different.	🖐
INITialize[:IMMediate] **Remark:** Handling of INITialize and TRIGger is different.	🖐

Device-specific functions	
Command syntax	**Status**
INITialize:MODE value INITialize:MODE? **Remark:** Handling of INITialize and TRIGger is different.	✋
INITialize:STATe value INITialize:STATe? **Remark:** Handling of INITialize and TRIGger is different.	✋
LFSource[:FREQuency] value LFSource[:FREQuency]?	✓
LFSource[:FREQuency]:STEP[:INCRement] value LFSource[:FREQuency]:STEP[:INCRement]?	✓
LFSource:LEVel value LFSource:LEVel?	✓
LFSource:LEVel:STEP[:INCRement] value LFSource:LEVel:STEP[:INCRement]?	✓
LFSource:STATe value LFSource:STATe?	✓
LFSource:WAVeform value LFSource:WAVeform ? **Remark:** SINE is supported.	✋
MODulation[:STATe] value MODulation[:STATe]?	✓
PHASe[:ADJust] value PHASe[:ADJust]?	✓
PHASe[:ADJust]:STEP[:INCRement] value PHASe[:ADJust]:STEP[:INCRement]?	✓
PHASe:REFerence	✓
[SOURce:]PM:COUPling value [SOURce:]PM:COUPling? **Remark:** AC, DC are supported.	✋
[SOURce:]PM[:DEViation] value [SOURce:]PM[:DEViation]?	✓
[SOURce:]PM[:DEViation]:STEP[:INCRement] value [SOURce:]PM[:DEViation]:STEP[:INCRement]?	✓

Device-specific functions	
Command syntax	**Status**
[SOURce:]PM:FREQuency value [SOURce:]PM:FREQuency?	✔
[SOURce:]PM:FREQuency:STEP[:INCRement] value [SOURce:]PM:FREQuency:STEP[:INCRement]?	✔
[SOURce:]PM:SOURce value [SOURce:]PM:SOURce?	✔
[SOURce:]PM:STATe value [SOURce:]PM:STATe?	✔
PULSe:DELay value PULSe:DELay?	✔
PULSe:DELay:STEP[:INCRement] value PULSe:DELay:STEP[:INCRement]?	✔
PULSe:FREQuency value PULSe:FREQuency?	✔
PULSe:FREQuency:STEP[:INCRement] value PULSe:FREQuency:STEP[:INCRement]?	✔
PULSe:SLOPe value PULSe:SLOPe ? **Remark:** POSitive, NEGative are supported.	🖐
PULSe:SOURce value PULSe:SOURce? **Remark:** INTernal, EXTernal are supported.	🖐
PULSe[:STATe] value PULSe[:STATe]?	✔
PULSe:WIDTh value PULSe:WIDTh?	✔
PULSe:WIDTh:STEP[:INCRement] value PULSe:WIDTh:STEP[:INCRement]?	✔
ROSCillator:CALibration value ROSCillator:CALibration?	✔
ROSCillator:CALibration:STEP[:INCRement] value ROSCillator:CALibration:STEP[:INCRement]?	✔
ROSCillator:SOURce value ROSCillator:SOURce?	✔

Device-specific functions	
Command syntax	**Status**
SWEep[:FREQuency]:MODE value SWEep[:FREQuency]:MODE?	✓
SWEep[:FREQuency]:SPACing value SWEep[:FREQuency]:SPACing?	✓
SWEep[:FREQuency]:TIME value SWEep[:FREQuency]:TIME? **Remark:** Numeric, MAXimum, MINimum are supported. Numeric items 1-2-5-10 are supported. Sweep time is divided into dwell time and count.	🖐
SWEep[:FREQuency]:TIME:STEP[:INCRement]?	✓
SWEep[:FREQuency]:TIME:STEP:MODE?	✓
SYSTem:ERRor? **Remark:** Returns the original error message of the Rohde & Schwarz signal generator.	🖐
TSWeep	✓

10 Emulating the Hewlett-Packard 8647/8648

The following tables show the current implementation status of each command. Commands not shown in these tables are not supported.

IEEE488.2 functions	
Command syntax	**Status**
*CAL?	✔
*CLS	✔
*ESE value *ESE?	✔
*ESR?	✔
*IDN? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	🖐
*OPC *OPC? Wait for/query completion of command.	✔
*OPT? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	🖐
*RCL value	✔
*RST	✔
*SAV value	✔
*SRE value *SRE?	✔
*STB?	✔
*TST?	✔
*WAI	✔

Device-specific functions	
Command syntax	**Status**
[SOURce:]AM[:DEPTh] value [SOURce:]AM[:DEPTh]?	✔
[SOURce:]AM:EXTernal:COUPling value [SOURce:]AM:EXTernal:COUPling?	✔

Device-specific functions	
Command syntax	**Status**
[SOURce:]AM:INTernal[1]:FREQuency value [SOURce:]AM:INTernal[1]:FREQuency?	✓
[SOURce:]AM:INTernal[1]:FUNCtion:SHAPe value [SOURce:]AM:INTernal[1]:FUNCtion:SHAPe? **Remark:** SINe is supported.	🖐
[SOURce:]AM:SOURce value [SOURce:]AM:SOURce? **Remark:** INTernal[1], EXTernal are supported.	🖐
[SOURce:]AM:STATe value [SOURce:]AM:STATe?	✓
[SOURce:]CAL:DCFM **Remark:** The elimination process takes about 10 s.	🖐
[SOURce:]FM[:DEViation] value [SOURce:]FM[:DEViation]?	✓
[SOURce:]FM:EXTernal:COUPling value [SOURce:]FM:EXTernal:COUPling?	✓
[SOURce:]FM:INTernal[1]:FREQuency value [SOURce:]FM:INTernal[1]:FREQuency?	✓
[SOURce:]FM:INTernal[1]:FUNCtion:SHAPe value. [SOURce:]FM:INTernal[1]:FUNCtion:SHAPe? **Remark:** SINe is supported.	🖐
[SOURce:]FM:SOURce value [SOURce:]FM:SOURce? **Remark:** INTernal[1], EXTernal are supported.	🖐
[SOURce:]FM:STATe value [SOURce:]FM:STATe?	✓
[SOURce:]FREQuency[:CW	:FIXed] value [SOURce:]FREQuency[:CW	:FIXed]?	✓
[SOURce:]FREQuency:REFerence value [SOURce:]FREQuency:REFerence?	✓
[SOURce:]FREQuency:REFerence:STATe value [SOURce:]FREQuency:REFerence:STATe?	✓

Device-specific functions	
Command syntax	**Status**
OUTPut[:STATe] value OUTPut[:STATe]?	✓
[SOURce:]PM[:DEViation] value [SOURce:]PM[:DEViation]?	✓
[SOURce:]PM:EXTernal:COUPling value [SOURce:]PM:EXTernal:COUPling?	✓
[SOURce:]PM:INTernal[1]:FREQuency value [SOURce:]PM:INTernal[1]:FREQuency?	✓
[SOURce:]PM:INTernal[1]:FUNCtion:SHAPe value [SOURce:]PM:INTernal[1]:FUNCtion:SHAPe? **Remark:** SINe is supported.	🖑
[SOURce:]PM:SOURce value [SOURce:]PM:SOURce? **Remark:** INTernal[1], EXTernal are supported.	🖑
[SOURce:]PM:STATe value [SOURce:]PM:STATe?	✓
[SOURce:]POWer:ATTenuation:AUTO value [SOURce:]POWer:ATTenuation:AUTO?	✓
[SOURce:]POWer[:LEVel][:IMMediate][:AMPLitude] value [SOURce:]POWer[:LEVel][:IMMediate][:AMPLitude]?	✓
[SOURce:]POWer:REFerence value [SOURce:]POWer:REFerence?	✓
[SOURce:]POWer:REFerence:STATe value [SOURce:]POWer:REFerence:STATe?	✓
[SOURce:]PULM:STATe value [SOURce:]PULM:STATe?	✓
STATus:OPERation[:EVENt]?	✓
STATus: OPERation:CONDition?	✓
STATus: OPERation:ENABle value STATus: OPERation:ENABle?	✓
STATus:QUEStionalbe[:EVENt]?	✓
STATus: QUEStionalbe:CONDition?	✓
STATus: QUEStionalbe:ENABle value STATus: QUEStionalbe:ENABle?	✓

Device-specific functions	
Command syntax	**Status**
SYSTem:ERRor? **Remark:** Returns the original error message of the Rohde & Schwarz signal generator.	🖐
SYSTem:LANGuage value SYSTem:LANGuage? **Remark:** On select, only "EXIT" is supported (to return to the instrument's native SCPI language). On query, always "SCPI" is returned.	🖐
SYSTem:VERSion?	✓

11 Emulating the Hewlett-Packard 8656/8657

The following tables show the current implementation status of each command. Commands not shown in these tables are not supported.

Device-specific functions	
Command syntax	**Status**
AM value	✔
AO value	✔
AP value **Remark:** Units DM, DF, DBVL, DBMV, DBUV, DBEMVL, DBEMMV, DBEMUV, VL, MV, UV, EMVL, EMMV, EMUV are supported.	✋
DN	✔
FM value	✔
FR value	✔
IP	✔
IS numeric	✔
PD	✔
PI	✔
PF	✔
PM	✔
R0	✔
R1	✔
R2	✔
R3	✔
R5	✔
RC value **Remark:** The contents of the registers 0 to 9 are NOT persistent and are lost when power is switched off or when leaving the emulation mode.	✋
RL value **Remark:** The contents of the registers 0 to 9 are NOT persistent and are lost when power is switched off or when leaving the emulation mode.	✋
RP	✔
S1	✔
S2	✔

Device-specific functions	
Command syntax	**Status**
S3	✔
S4	✔
S5 value	✔
UP	✔
SV value **Remark:** The contents of the registers 0 to 9 are NOT persistent and are lost when power is switched off or when leaving the emulation mode.	✋
ST value **Remark:** The contents of the registers 0 to 9 are NOT persistent and are lost when power is switched off or when leaving the emulation mode.	✋

12 Emulating the Panasonic 8303

The following tables show the current implementation status of each command. Commands not shown in these tables are not supported.

Device-specific functions	
Command syntax	**Status**
AFA value	✔
AFF value	✔
AM value	✔
AP value	✔
AS	◯
DM	◯
DR	◯
EA value	✔
EF value	✔
EM	◯
FM value	✔
FR value	✔
LE value	✔
MO	✔
MS	◯
NT value	✔
PL	◯
PR	◯
P1	◯
P2	◯
QG	✔
RC value **Remark:** The contents of the registers 0 to 99 are NOT persistent and are lost when power is switched off or when leaving the emulation mode.	🖐
TM	◯
TO value	✔

Device-specific functions	
Command syntax	**Status**
ST value **Remark:** The contents of the registers 0 to 99 are NOT persistent and are lost when power is switched off or when leaving the emulation mode.	🖑

13 Emulating the R&S®SML

The following tables show the current implementation status of each command. Commands not shown in these tables are not supported.

IEEE488.2 functions	
Command syntax	**Status**
*CAL?	✔
*CLS	✔
*ESE value *ESE?	✔
*ESR?	✔
*IDN? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	✋
*OPC *OPC? Wait for/query completion of command.	✔
*OPT? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	✋
*RCL value	✔
*RST	✔
*SAV value	✔
*SRE value *SRE?	✔
*STB?	✔
*TST?	✔
*WAI	✔

Device-specific functions	
Command syntax	**Status**
ABORt[:SWEep]	✔
CALibration[:ALL]?	✔
CALibration:FMOFfset[:MEASure]?	✔
CALibration:ROSCillator[:DATA]? Value	✔
DIAGnostic:INFO:CCOunt:POWer?	✔

Device-specific functions	
Command syntax	**Status**
DIAGnostic:INFO:OTIMe?	✓
DIAGnostic:INFO:SDATe?	✓
DISPlay:ANNotation[:ALL] value DISPlay:ANNotation[:ALL]?	✓
DISPlay:ANNotation:AMPLitude value DISPlay:ANNotation:AMPLitude?	✓
DISPlay:ANNotation:FREQuency value DISPlay:ANNotation:FREQuency?	✓
OUTPut1:AMODe value OUTPut1:AMODe?	✓
OUTPut1[:STATe] value OUTPut1[:STATe]?	✓
OUTPut2[:STATe] value OUTPut2[:STATe]?	✓
OUTPut1[:STATe]:PON value OUTPut1[:STATe]:PON?	✓
OUTPut2:VOLTage value OUTPut2:VOLTage?	✓
[SOURce:]AM[:DEPTh] value [SOURce:]AM[:DEPTh]?	✓
[SOURce:]AM:EXTernal:COUPling value [SOURce:]AM:EXTernal:COUPling?	✓
[SOURce:]AM:INTernal:FREQuency value [SOURce:]AM:INTernal:FREQuency? value	✓
[SOURce:]AM:SOURce value [SOURce:]AM:SOURce?	✓
[SOURce:]AM:STATe value [SOURce:]AM:STATe?	✓
[SOURce:]CORRection[:STATe] value [SOURce:]CORRection[:STATe]?	✓
[SOURce:]CORRection:CSET:CATalog?	✓
[SOURce:]CORRection:CSET:FREE?	✓
[SOURce:]CORRection:CSET[:SELect] value [SOURce:]CORRection:CSET[:SELect]?	✓
[SOURce:]CORRection:CSET:DATA:FREQuency value [SOURce:]CORRection:CSET:DATA:FREQuency?	✓

Device-specific functions	
Command syntax	**Status**
[SOURce:]CORRection:CSET:DATA:FREQuency:POINts?	✓
[SOURce:]CORRection:CSET:DATA:POWer value [SOURce:]CORRection:CSET:DATA:POWer?	✓
[SOURce:]CORRection:CSET:DATA:POWer:POINts?	✓
[SOURce:]CORRection:CSET:DELete value	✓
[SOURce:]CORRection:CSET:DELete:ALL	✓
[SOURce:]FM[:DEViation] value [SOURce:]FM[:DEViation]? **Remark:** Numeric, MINimum are supported.	✋
[SOURce:]FM:EXTernal:COUPling value [SOURce:]FM:EXTernal:COUPling?	✓
[SOURce:]FM:INTernal:FREQuency value [SOURce:]FM:INTernal:FREQuency?	✓
[SOURce:]FM:SOURce value [SOURce:]FM:SOURce?	✓
[SOURce:]FM:STATe value [SOURce:]FM:STATe?	✓
[:SOURce:]FREQuency:CENTer value [:SOURce:]FREQuency:CENTer?	✓
[:SOURce:]FREQuency[:CW | :FIXed] value [:SOURce:]FREQuency[:CW | :FIXed]?	✓
[:SOURce:]FREQuency:RCL value [:SOURce:]FREQuency:RCL?	✓
[:SOURce:]FREQuency:MANual value Sets RF frequency in sweep mode. **Remark:** Sets and limits actual RF frequency to sweep range.	✋
[:SOURce:]FREQuency:MODE value [:SOURce:]FREQuency:MODE?	✓
[:SOURce:]FREQuency:OFFSet value [:SOURce:]FREQuency:OFFSet?	✓
[:SOURce:]FREQuency:SPAN value [:SOURce:]FREQuency:SPAN?	✓
[:SOURce:]FREQuency:STARt value [:SOURce:]FREQuency:STARt?	✓

Device-specific functions	
Command syntax	**Status**
[:SOURce:]FREQuency:STOP value [:SOURce:]FREQuency:STOP?	✔
[:SOURce:]FREQuency:STEP[:INCRement] value [:SOURce:]FREQuency:STEP[:INCRement]?	✔
[:SOURce:]MODulation[:ALL]:STATe value [:SOURce:]MODulation[:ALL]:STATe?	✔
[:SOURce:]PHASe value [:SOURce:]PHASe?	✔
[:SOURce:]PHASe:REFerence	✔
[:SOURce:]PHASe:STEP value [:SOURce:]PHASe:STEP?	✔
[:SOURce:]PM[:DEViation] value [:SOURce:]PM[:DEViation]? Sets/queries the modulation deviation. **Remark:** Numeric, MINimum are supported.	🖐
[:SOURce:]PM:EXTernal:COUPling value [:SOURce:]PM:EXTernal:COUPling?	✔
[:SOURce:]PM:INTernal:FREQuency value [:SOURce:]PM:INTernal:FREQuency?	✔
[:SOURce:]PM:SOURce value [:SOURce:]PM:SOURce?	✔
[:SOURce:]PM:STATe value [:SOURce:]PM:STATe?	✔
[:SOURce:]POWer:ALC:STATe value [:SOURce:]POWer:ALC:STATe?	✔
[:SOURce:]POWer[:LEVel][:IMMediate][:AMPLitude] value [:SOURce:]POWer[:LEVel][:IMMediate][:AMPLitude]?	✔
[:SOURce:]POWer[:LEVel][:IMMediate]:OFFSet value [:SOURce:]POWer[:LEVel][:IMMediate]:OFFSet?	✔
[:SOURce:]POWer:LIMit[:AMPLitude] value [:SOURce:]POWer:LIMit[:AMPLitude]?	✔
[:SOURce:]POWer:MANual value Sets the RF level in sweep mode. **Remark:** Sets and limits actual RF level to sweep range.	🖐

Device-specific functions	
Command syntax	**Status**
[:SOURce:]POWer:MODE value [:SOURce:]POWer:MODE?	✓
[:SOURce:]POWer:RCL value [:SOURce:]POWer:RCL?	✓
[:SOURce:]POWer:STARt value [:SOURce:]POWer:STARt?	✓
[:SOURce:]POWer:STEP[:INCRement] value [:SOURce:]POWer:STEP[:INCRement]?	✓
[:SOURce:]POWer:STOP value [:SOURce:]POWer:STOP?	✓
[:SOURce:]PULM:POLarity? **Remark:** In external mode only.	🖐
[:SOURce:]PULM:SOURce value [:SOURce:]PULM:SOURce?	✓
[:SOURce:]PULM:STATe value [:SOURce:]PULM:STATe?	✓
[:SOURce:]PULSe:DELay value [:SOURce:]PULSe:DELay?	✓
[:SOURce:]PULSe:DOUBle:DELay value [:SOURce:]PULSe:DOUBle:DELay?	✓
[:SOURce:]PULSe:DOUBle:STATe value [:SOURce:]PULSe:DOUBle:STATe?	✓
[:SOURce:]PULSe:PERiod value [:SOURce:]PULSe:PERiod?	✓
[:SOURce:]PULSe:WIDTh value [:SOURce:]PULSe:WIDTh	✓
[:SOURce:]ROSCillator[:INTernal]:ADJust[:STATe] value [:SOURce:]ROSCillator[:INTernal]:ADJust[:STATe]?	✓
[:SOURce:]ROSCillator[:INTernal]:ADJust:VALue value [:SOURce:]ROSCillator[:INTernal]:ADJust:VALue?	✓
[:SOURce:]ROSCillator:SOURce value [:SOURce:]ROSCillator:SOURce?	✓
[SOURce:]SWEep[:FREQuency]:DWELl value [SOURce:]SWEep[:FREQuency]:DWELl?	✓

Device-specific functions	
Command syntax	**Status**
[SOURce:]SWEep[:FREQuency]:MODE value [SOURce:]SWEep[:FREQuency]:MODE? **Remark:** AUTO, MAN are supported.	🖐
[SOURce:]SWEep[:FREQuency]:RUNNing?	✔
[SOURce:]SWEep[:FREQuency]:SPACing value [SOURce:]SWEep[:FREQuency]:SPACing?	✔
[SOURce:]SWEep[:FREQuency]:STEP[:LINEar] value [SOURce:]SWEep[:FREQuency]:STEP[:LINEar]?	✔
[SOURce:]SWEep[:FREQuency]:STEP:LOGarithmic value [SOURce:]SWEep[:FREQuency]:STEP:LOGarithmic?	✔
[SOURce:]SWEep:POWer:DWELl value [SOURce:]SWEep:POWer:DWELl?	✔
[SOURce:]SWEep:POWer:MODE value [SOURce:]SWEep:POWer:MODE? **Remark:** AUTO, MAN are supported.	🖐
[SOURce:]SWEep:POWer:RUNNing?	✔
[SOURce:]SWEep:POWer:SPACing value [SOURce:]SWEep:POWer:SPACing?	✔
[SOURce:]SWEep:POWer:STEP[:LOGarithmic] value [SOURce:]SWEep:POWer:STEP[:LOGarithmic]?	✔
SOURce2:FREQuency[:CW | :FIXed] value SOURce2:FREQuency[:CW | :FIXed]?	✔
SOURce2:FREQuency:MANual value Sets LF frequency in sweep mode. **Remark:** Sets and limits actual LF frequency to sweep range.	🖐
SOURce2:FREQuency:MODE value SOURce2:FREQuency:MODE?	✔
SOURce2:FREQuency:STARt value SOURce2:FREQuency:STARt?	✔
SOURce2:FREQuency:STOP value SOURce2:FREQuency:STOP?	✔
SOURce2:SWEep[:FREQuency]:DWELl value SOURce2:SWEep[:FREQuency]:DWELl?	✔

Device-specific functions	
Command syntax	**Status**
SOURce2:SWEep[:FREQuency]:MODE value SOURce2:SWEep[:FREQuency]:MODE? **Remark:** AUTO, MAN are supported.	🖐
SOURce2:SWEep[:FREQuency]:RUNNing?	✔
SOURce2:SWEep[:FREQuency]:SPACing value SOURce2:SWEep[:FREQuency]:SPACing?	✔
SOURce2:SWEep[:FREQuency]:STEP[:LINear] value SOURce2:SWEep[:FREQuency]:STEP[:LINear]?	✔
SOURce2:SWEep[:FREQuency]:STEP:LOGarithmic value SOURce2:SWEep[:FREQuency]:STEP:LOGarithmic?	✔
STATus:OPERation:CONDition?	✔
STATus:OPERation:ENABle value STATus:OPERation:ENABle?	✔
STATus:OPERation[:EVENt]?	✔
STATus:OPERation:NTRansition value STATus:OPERation: NTRansition?	✔
STATus:OPERation:PTRansition value STATus:OPERation: PTRansition?	✔
STATus:PRESet	✔
STATus:QUEStionable:CONDition?	✔
STATus:QUEStionable:ENABle value STATus:QUEStionable:ENABle?	✔
STATus:QUEStionable[:EVENt]?	✔
STATus:QUEStionable:NTRansition value STATus:QUEStionable: NTRansition?	✔
STATus:QUEStionable:PTRansition value STATus:QUEStionable: PTRansition?	✔
STATus:QUEue[:NEXT]?	✔
SYSTem:COMMunicate:GPIB[:SELF]:ADDRess value SYSTem:COMMunicate:GPIB[:SELF]:ADDRess?	✔
SYSTem:COMMunicate:SERial:BAUD value SYSTem:COMMunicate:SERial:BAUD?	✔
SYSTem:COMMunicate:SERial:SBITs value SYSTem:COMMunicate:SERial:SBITs?	✔

Device-specific functions	
Command syntax	**Status**
SYSTem:COMMunicate:SERial:PARity value SYSTem:COMMunicate:SERial:PARity?	✓
SYSTem:DISPlay:UPDate[:STATe] value SYSTem:DISPlay:UPDate[:STATe]?	✓
SYSTem:ERRor? **Remark:** Returns the original error message of the Rohde & Schwarz signal generator.	🖐
SYSTem:PRESet **Remark:** Implementation is identical to common command *RST.	🖐
SYSTem:SECurity[:STATe] value SYSTem:SECurity[:STATe]?	✓
SYSTem:SERRor?	✓
SYSTem:VERSion?	✓
TRIGger[1] [:SWEep][:IMMediate]	✓
TRIGger2[:SWEep][:IMMediate]	✓
TRIGger[1] [:SWEep]:SOURce value TRIGger[1][:SWEep]:SOURce?	✓
TRIGger2[:SWEep]:SOURce value TRIGger2[:SWEep]:SOURce?	✓
TRIGger:PULSe:EGATed:POLarity value TRIGger:PULSe:EGATed:POLarity?	✓
TRIGger:PULSe:SOURce value TRIGger:PULSe:SOURce? **Remark:** AUTO, EGATe, EXTernal are supported.	🖐
TRIGger:PULSe:SLOPe value TRIGger:PULSe:SLOPe?	✓
UNIT:POWer value UNIT:POWer?	✓

14 Emulating the R&S®SME, SMT

The following tables show the current implementation status of each command. Commands not shown in these tables are not supported.

IEEE488.2 functions	
Command syntax	**Status**
*CAL?	✔
*CLS	✔
*ESE value *ESE?	✔
*ESR?	✔
*IDN? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	✋
*OPC *OPC?.	✔
*OPT? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	✋
*RCL value	✔
*RST	✔
*SAV value	✔
*SRE value *SRE?	✔
*STB?	✔
*TST?	✔
*WAI	✔

Device-specific functions	
Command syntax	**Status**
ABORt[:SWEep]	✔
CALibration:FM:OFfset?	✔
CALibration:LEVel:STATe value CALibration:LEVel:STATe?	✔
CALibration:ROSCillator[:DATA] value CALibration:ROSCillator[:DATA]?	✔

Device-specific functions	
Command syntax	**Status**
DIAGnostic:INFO:CCOunt:POWer?	✓
DIAGnostic:INFO:OTIMe?	✓
DIAGnostic:INFO:SDATe?	✓
DISPlay:ANNotation[:ALL] value DISPlay:ANNotation[:ALL]?	✓
DISPlay:ANNotation:AMPLitude value DISPlay:ANNotation:AMPLitude?	✓
DISPlay:ANNotation:FREQuency value DISPlay:ANNotation:FREQuency?	✓
FORMat[:DATA] value FORMat[:DATA]?	✓
MEMory:NSTates?	✓
OUTPut:AMODe value OUTPut:AMODe?	✓
OUTPut[:STATe] value OUTPut[:STATe]?	✓
OUTPut2[:STATe] value OUTPut2[:STATe]?	✓
OUTPut[:STATe]:PON value OUTPut[:STATe]:PON?	✓
OUTPut2:VOLTage value OUTPut2:VOLTage?	✓
[SOURce:]AM[:DEPTh] value [SOURce:]AM[:DEPTh]?	✓
[SOURce:]AM:EXTernal:COUPling value [SOURce:]AM:EXTernal:COUPling?	✓
[SOURce:]AM:INTernal:FREQuency value [SOURce:]AM:INTernal:FREQuency? value	✓
[SOURce:]AM:SOURce value [SOURce:]AM:SOURce? **Remark:** INTernal1, EXTernal1 are supported.	✋
[SOURce:]AM:STATe value [SOURce:]AM:STATe?	✓
[SOURce:]CORRection:CSET:CATalog?	✓
[SOURce:]CORRection:CSET:FREE?	✓

Device-specific functions	
Command syntax	**Status**
[SOURce:]CORRection:CSET[:SELect] value [SOURce:]CORRection:CSET[:SELect]?	✔
[SOURce:]CORRection:CSET:DATA:FREQuency value [SOURce:]CORRection:CSET:DATA:FREQuency?	✔
[SOURce:]CORRection:CSET:DATA:FREQuency:POINts?	✔
[SOURce:]CORRection:CSET:DATA:POWer value [SOURce:]CORRection:CSET:DATA:POWer?	✔
[SOURce:]CORRection:CSET:DATA:POWer:POINts?	✔
[SOURce:]CORRection:CSET:DELete value	✔
[SOURce:]CORRection:CSET:DELete:ALL	✔
[SOURce:]CORRection[:STATe] value [SOURce:]CORRection[:STATe]?	✔
[SOURce:]FM[1][:DEViation] value [SOURce:]FM[1] [:DEViation]? **Remark:** Numeric, MINimum are supported.	✋
[SOURce:]FM[1]:EXTernal:COUPling value [SOURce:]FM[1]:EXTernal:COUPling?	✔
[SOURce:]FM[1]:INTernal:FREQuency value [SOURce:]FM[1]:INTernal:FREQuency?	✔
[SOURce:]FM[1]:SOURce value [SOURce:]FM[1]:SOURce? **Remark:** INTernal1, EXTernal1 are supported.	✋
[SOURce:]FM[1]:STATe value [SOURce:]FM[1]:STATe?	✔
[:SOURce:]FREQuency:CENTer value [:SOURce:]FREQuency:CENTer?	✔
[:SOURce:]FREQuency[:CW] value [:SOURce:]FREQuency[:CW]?	✔
[:SOURce:]FREQuency[:CW]:RCL value [:SOURce:]FREQuency[:CW]:RCL?	✔
[:SOURce:]FREQuency:FIXed value [:SOURce:]FREQuency:FIXed?	✔
[:SOURce:]FREQuency:FIXed:RCL value [:SOURce:]FREQuency:FIXed:RCL?	✔

Device-specific functions	
Command syntax	**Status**
[:SOURce:]FREQuency:MANual value Sets RF frequency in sweep mode. **Remark:** Sets and limits actual RF frequency to sweep range.	🖐
[:SOURce:]FREQuency:MODE value [:SOURce:]FREQuency:MODE?	✔
[:SOURce:]FREQuency:OFFSet value [:SOURce:]FREQuency:OFFSet?	✔
[:SOURce:]FREQuency:SPAN value [:SOURce:]FREQuency:SPAN?	✔
[:SOURce:]FREQuency:STARt value [:SOURce:]FREQuency:STARt?	✔
[:SOURce:]FREQuency:STOP value [:SOURce:]FREQuency:STOP?	✔
[:SOURce:]FREQuency:STEP[:INCRement] value [:SOURce:]FREQuency:STEP[:INCRement]?	✔
[:SOURce:]PHASe[:ADJust] value [:SOURce:]PHASe[:ADJust]?	✔
[:SOURce:]PHASe:REFerence	✔
[:SOURce:]PM[1][:DEViation] value [:SOURce:]PM[1] [:DEViation]? **Remark:** Numeric, MINimum are supported.	🖐
[:SOURce:]PM[1]:EXTernal:COUPling value [:SOURce:]PM[1]:EXTernal:COUPling?	✔
[:SOURce:]PM[1]:INTernal:FREQuency value [:SOURce:]PM[1]:INTernal:FREQuency?	✔
[:SOURce:]PM[1]:SOURce value [:SOURce:]PM[1]:SOURce? **Remark:** INTernal1, EXTernal1 are supported.	🖐
[:SOURce:]PM[1]:STATe value [:SOURce:]PM[1]:STATe?	✔
[:SOURce:]POWer:ALC:STATe value [:SOURce:]POWer:ALC:STATe?	✔
[:SOURce:]POWer[:LEVel][:IMMediate][:AMPLitude] value [:SOURce:]POWer[:LEVel][:IMMediate][:AMPLitude]?	✔

Device-specific functions	
Command syntax	**Status**
[:SOURce:]POWer[:LEVel][:IMMediate]:OFFSet value [:SOURce:]POWer[:LEVel][:IMMediate]:OFFSet?	✓
[:SOURce:]POWer[:LEVel][:IMMediate]:RCL value [:SOURce:]POWer[:LEVel][:IMMediate]:RCL?	✓
[:SOURce:]POWer:LIMit[:AMPLitude] value [:SOURce:]POWer:LIMit[:AMPLitude]?	✓
[:SOURce:]POWer:MANual value **Remark:** Sets and limits actual RF level to sweep range.	🖐
[:SOURce:]POWer:MODE value [:SOURce:]POWer:MODE?	✓
[:SOURce:]POWer:STARt value [:SOURce:]POWer:STARt?	✓
[:SOURce:]POWer:STEP[:INCRement] value [:SOURce:]POWer:STEP[:INCRement]?	✓
[:SOURce:]POWer:STOP value [:SOURce:]POWer:STOP?	✓
[:SOURce:]PULM:EXTernal: IMPedance value [:SOURce:]PULM:EXTernal:IMPedance?	✓
[:SOURce:]PULM:INTernal:FREQuency value [:SOURce:]PULM:INTernal:FREQuency?	✓
[:SOURce:]PULM:POLarity? **Remark:** In external mode only.	🖐
[:SOURce:]PULM:SOURce value [:SOURce:]PULM:SOURce?	✓
[:SOURce:]PULM:STATe value [:SOURce:]PULM:STATe?	✓
[:SOURce:]PULSe:DELay value [:SOURce:]PULSe:DELay?	✓
[:SOURce:]PULSe:DOUBle:DELay value [:SOURce:]PULSe:DOUBle:DELay?	✓
[:SOURce:]PULSe:DOUBle:STATe value [:SOURce:]PULSe:DOUBle:STATe?	✓
[:SOURce:]PULSe:PERiod value [:SOURce:]PULSe:PERiod?	✓

Device-specific functions	
Command syntax	**Status**
[:SOURce:]PULSe:WIDTh value [:SOURce:]PULSe:WIDTh	✓
[:SOURce:]ROSCillator[:INTernal]:ADJust[:STATe] value [:SOURce:]ROSCillator[:INTernal]:ADJust[:STATe]?	✓
[:SOURce:]ROSCillator[:INTernal]:ADJust:VALue value [:SOURce:]ROSCillator[:INTernal]:ADJust:VALue?	✓
[:SOURce:]ROSCillator:SOURce value [:SOURce:]ROSCillator:SOURce?	✓
[SOURce:]SWEep[:FREQuency]:DWELl value [SOURce:]SWEep[:FREQuency]:DWELl?	✓
[SOURce:]SWEep[:FREQuency]:MODE value [SOURce:]SWEep[:FREQuency]:MODE? **Remark:** AUTO, MAN are supported.	🖐
[SOURce:]SWEep[:FREQuency]:POINts value [SOURce:]SWEep[:FREQuency]: POINts?	✓
[SOURce:]SWEep[:FREQuency]:SPACing value [SOURce:]SWEep[:FREQuency]:SPACing?	✓
[SOURce:]SWEep[:FREQuency]:STEP[:LINEar] value [SOURce:]SWEep[:FREQuency]:STEP[:LINEar]?	✓
[SOURce:]SWEep[:FREQuency]:STEP:LOGarithmic value [SOURce:]SWEep[:FREQuency]:STEP:LOGarithmic?	✓
[SOURce:]SWEep:POWer:DWELl value [SOURce:]SWEep:POWer:DWELl?	✓
[SOURce:]SWEep:POWer:MODE value [SOURce:]SWEep:POWer:MODE? **Remark:** AUTO, MAN are supported.	🖐
[SOURce:]SWEep:POWer:POINts value [SOURce:]SWEep:POWer:POINts?	✓
[SOURce:]SWEep:POWer:SPACing value [SOURce:]SWEep:POWer:SPACing?	✓
[SOURce:]SWEep:POWer:STEP[:LOGarithmic] value [SOURce:]SWEep:POWer:STEP[:LOGarithmic]?	✓
SOURce2:FREQuency[:CW] value SOURce2:FREQuency[:CW]?	✓

Device-specific functions	
Command syntax	**Status**
SOURce2:FREQuency:FIXed value SOURce2:FREQuency:FIXed?	✓
SOURce2:FREQuency:MANual value **Remark:** Sets and limits actual LF frequency to sweep range.	🖐
SOURce2:FREQuency:MODE value SOURce2:FREQuency:MODE?	✓
SOURce2:FREQuency:STARt value SOURce2:FREQuency:STARt?	✓
SOURce2:FREQuency:STOP value SOURce2:FREQuency:STOP?	✓
SOURce2:FUNCtion value SOURce2: FUNCtion? **Remark:** SINusoid, SQUare are supported.	🖐
SOURce2:SWEep[:FREQuency]:DWELl value SOURce2:SWEep[:FREQuency]:DWELl?	✓
SOURce2:SWEep[:FREQuency]:MODE value SOURce2:SWEep[:FREQuency]:MODE? **Remark:** AUTO, MAN are supported.	🖐
SOURce2:SWEep[:FREQuency]:POINts value SOURce2:SWEep[:FREQuency]: POINts?	✓
SOURce2:SWEep[:FREQuency]:SPACing value SOURce2:SWEep[:FREQuency]:SPACing?	✓
SOURce2:SWEep[:FREQuency]:STEP[:LINear] value SOURce2:SWEep[:FREQuency]:STEP[:LINear]?	✓
SOURce2:SWEep[:FREQuency]:STEP:LOGarithmic value SOURce2:SWEep[:FREQuency]:STEP:LOGarithmic?	✓
STATus:OPERation:CONDition?	✓
STATus:OPERation:ENABle value STATus:OPERation:ENABle?	✓
STATus:OPERation[:EVENt]?	✓
STATus:OPERation:NTRansition value STATus:OPERation: NTRansition?	✓
STATus:OPERation:PTRansition value STATus:OPERation: PTRansition?	✓

Device-specific functions	
Command syntax	**Status**
STATus:PRESet	✔
STATus:QUEStionable:CONDition?	✔
STATus:QUEStionable:ENABle value STATus:QUEStionable:ENABle?	✔
STATus:QUEStionable[:EVENt]?	✔
STATus:QUEStionable:NTRansition value STATus:QUEStionable: NTRansition?	✔
STATus:QUEStionable:PTRansition value STATus:QUEStionable: PTRansition?	✔
STATus:QUEue[:NEXT]?	✔
SYSTem:BEEPer:STATe value SYSTem:BEEPer:STATe?	✔
SYSTem:COMMunicate:GPIB[:SELF]:ADDRess value SYSTem:COMMunicate:GPIB[:SELF]:ADDRess?	✔
SYSTem:COMMunicate:SERial:BAUD value SYSTem:COMMunicate:SERial:BAUD? **Remark:** 2400, 4800, 9600, 19200, 38400, 57600, 115200 are supported.	✋
SYSTem:ERRor? **Remark:** Returns the original error message of the Rohde & Schwarz signal generator.	✋
SYSTem:KLOCk value SYSTem:KLOCk?	✔
SYSTem:PRESet **Remark:** Implementation is identical to common command *RST.	✋
SYSTem:SECurity[:STATe] value SYSTem:SECurity[:STATe]?	✔
SYSTem:SERRor?	✔
SYSTem:VERSion?	✔
TRIGger:PULSe:SOURce value TRIGger:PULSe:SOURce?	✔
TRIGger:PULSe:SLOPe value TRIGger:PULSe:SLOPe?	✔

Device-specific functions	
Command syntax	**Status**
TRIGger:SLOPe value TRIGger:SLOPe? **Remark:** NEGative, POSitve are supported.	🖐
TRIGger[:SWEep][:IMMediate]	✔
TRIGger2[:SWEep][:IMMediate]	✔
TRIGger[:SWEep]:SOURce value TRIGger[:SWEep]:SOURce?	✔
TRIGger2[:SWEep]:SOURce value TRIGger2[:SWEep]:SOURce?	✔
UNIT:ANGle value UNIT:ANGle?	✔
UNIT:POWer value UNIT:POWer?	✔

15 Emulating the R&S®SMY

The following tables show the current implementation status of each command. Commands not shown in these tables are not supported.

IEEE488.2 functions	
Command syntax	**Status**
*CLS	✔
*ESE value *ESE?	✔
*ESR?	✔
*IDN? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	✋
*OPC *OPC?	✔
*OPT? **Remark:** If the response does not match the requirements, a user-specific response to *IDN? and *OPT? can be applied on the instrument's front panel.	✋
*RCL value **Remark:** Only values in the range from 0 to 9 are supported.	✋
*RST	✔
*SAV value **Remark:** Only values in the range from 0 to 9 are supported.	✋
*SRE value *SRE?	✔
*STB?	✔
*TST?	✔
*WAI	✔

Device-specific functions	
Command syntax	**Status**
AF value	✔
AF:OFF	✔
AF:ON	✔

Device-specific functions	
Command syntax	**Status**
AF:Var_step value	✔
AF?	✔
AF:Var_step?	✔
ALc:Auto	○
ALc:Fixed	✔
ALc:NARrow	○
ALc:Normal	✔
ALc:WIDe	○
ALc?	✔
AM value	✔
AM:External:Ac	✔
AM:External:Ac value	✔
AM:External:Dc	✔
AM:External:Dc value	✔
AM:Internal	✔
AM:Internal value	✔
AM:OFF	✔
AM:Var_step value	✔
AM?	✔
AM:Var_step?	✔
ATtenuator:Fixed	✔
ATtenuator:Normal	✔
Attenuator?	✔
DECrement:AF	✔
DECrement:AM	✔
DECrement:FM	✔
DECrement:Level	✔
DECrement:PHM	✔
DECrement:RF	✔
FM value	✔
FM:External:Ac	✔
FM:External:Ac value	✔
FM:External:Dc	✔

Device-specific functions	
Command syntax	**Status**
FM:External:Dc value	✔
FM:Internal	✔
FM:Internal value	✔
FM:OFF	✔
FM:Var_step value	✔
FM?	✔
FM:Var_step?	✔
HEAder:OFF	✔
HEAder:ON	✔
INCrement:AF	✔
INCrement:AM	✔
INCrement:FM	✔
INCrement:Level	✔
INCrement:PHM	✔
INCrement:RF	✔
Level value	✔
Level:Emf value	✔
Level:OFF	✔
Level:ON	✔
Level:Var_step value	✔
Level?	✔
Level:Emf?	✔
Level:Var_step?	✔
PHM value	✔
PHM:External	✔
PHM:External value	✔
PHM:Internal	✔
PHM:Internal value	✔
PHM:OFF	✔
PHM:Var_step value	✔
PHM?	✔
PHM:Var_step?	✔
PReset	✔

Device-specific functions	
Command syntax	**Status**
REFerence_oscillator:External **Remark:** Only the first 12 characters are valid.	✋
REFerence_oscillator:Internal **Remark:** Only the first 12 characters are valid.	✋
REFerence_oscillator? **Remark:** Only the first 12 characters are valid.	✋
RF value	✔
RF:STArt value	✔
RF:STEp value	✔
RF:STOp value	✔
RF:Var_step value	✔
RF?	✔
RF:STArt?	✔
RF:STEp?	✔
RF:STOp?	✔
RF:Var_step?	✔
SWP:Auto	✔
SWP:OFF	✔
SWP:ON	✔
SWP:Reset	✔
SWP?	✔
Time[:RF_swp] value	✔
Time[:RF_swp]?	✔

16 References

[1]   Rohde & Schwarz, R&S®SMC100A Signal Generator Operating Manual

17 Additional Information

This application note is updated from time to time. Please visit the following website in order to download the latest version:

Application Notes about the R&S®SMC100A Signal Generator

18 Ordering Information

Please visit the following website for comprehensive ordering information for the SMC:

Ordering Information for the R&S®SMC100A Signal Generator

R&S®SMC100A	**Signal Generator**	1411.4002.02
R&S®SMC-B101	9 kHz to 1.1 GHz	1411.6505.02
R&S®SMC-B103	9 kHz to 3.2 GHz	1411.6605.02
R&S®SMC-B1	OCXO Reference Oscillator	1411.6705.02
R&S®SMC-K4	GPIB/IEEE 488 Interface	1411.3506.02

About Rohde & Schwarz

Rohde & Schwarz is an independent group of companies specializing in electronics. It is a leading supplier of solutions in the fields of test and measurement, broadcasting, radiomonitoring and radiolocation, as well as secure communications. Established more than 75 years ago, Rohde & Schwarz has a global presence and a dedicated service network in over 70 countries. Company headquarters are in Munich, Germany.

Environmental commitment

- Energy-efficient products
- Continuous improvement in environmental sustainability
- ISO 14001-certified environmental management system

Certified Quality System
ISO 9001

Regional contact

Europe, Africa, Middle East
+49 89 4129 12345
customersupport@rohde-schwarz.com

North America
1-888-TEST-RSA (1-888-837-8772)
customer.support@rsa.rohde-schwarz.com

Latin America
+1-410-910-7988
customersupport.la@rohde-schwarz.com

Asia/Pacific
+65 65 13 04 88
customersupport.asia@rohde-schwarz.com

China
+86-800-810-8228 /+86-400-650-5896
customersupport.china@rohde-schwarz.com

This application note and the supplied programs may only be used subject to the conditions of use set forth in the download area of the Rohde & Schwarz website.