SK4M Spectrum Analyzers

SCPI Command Reference

VERSION 1.4 (Graphit SK4M 2.3.9 or newer)
11 November 2019
Micran, Research & Production Company

Contents

1. SCPI Basics

1.1 Introduction

The Standard Commands for Programmable Instruments (SCPI) defines a set of standard programming commands for use by all SCPI compatible instruments. This section describes the general use of the SCPI language for the Micran instruments.

For additional information refer to the SCPI Consortium or IEEE (IEEE Standard 488.2).

1.2 Command Structure

All SCPI commands, except the common commands, are organized in a hierarchical structure similar to the inverted tree file structure used in most computers. The command keywords that correspond to the major instrument control functions are located at the top of the command tree. All SCPI commands, except the ABORt command, have one or more subcommands (keywords) associated with them to further define the instrument function to be controlled. The subcommand keywords may in turn also have one or more associated subcommands (keywords). Each subcommand level adds another layer to the command tree. The command keyword and its associated subcommand keywords form a portion of the command tree called a command subsystem. The :INITiate command subsystem is shown below.

```
:INITiate
        :CONTinuous
                <bool>
        :[:IMMediate]
```

1.3 Subsystems

Subsystem commands are distinguished by the colon (:). The colon is used at the beginning of a command statement and between keywords. For example:

```
:SYSTem:ERRor:COUNt?
```

"COUNt" is the query of the "ERRor" subsystem contained in the "SYSTem" top command tree.

1.4 Mnemonic Generation Rules

Each instrument-control header or keyword has both a long and a short form. A SCPI instrument shall accept only the exact short and the exact long forms. Sending a header that is not the short form, nor the complete long form to a SCPI instrument shall cause it to generate an error. For example:

:INPut:ATTenuation

command can be typed as:

:INP:ATT

Incorrect command notation listed below:

:INPU:ATTenuation

1.5 Letter case

Lowercase and uppercase letters are considered equivalent:

:INP:ATTenuation and :inP:AtT

1.6 Parameters

A typical command is made up of keywords prefixed with colons (:). The keywords are followed by parameters. There is a separating space (white space) between the command and its parameter. Few parameters should be separated using commas (','). Example:

:FREQ:LIST 1000 MHz, 2000 MHz, 3000 MHz, 4000 MHz

1.6.1 Numeric formats (<numeric>)

Numeric parameters are used in both common and subsystem commands. They accept all commonly used decimal representations of numbers including optional signs, decimal points, and scientific notation.

The following syntax conventions are used for numeric data parameters:

- <NR1> - a signed integer without a decimal point (implied radix point), e.g.: 12, +23, -656;
- <NR2> - a signed number with an explicit radix point, e.g.: 12.571;
- <NR3> - a scaled explicit decimal point numeric value with radix and exponent (e.g., floating point number), such as 12.451E4 (equals 124510).

Extended numeric parameters also include the following special parameters:

- DEFault - resets the parameter to its default value;
- UP - increments the parameter;
- DOWN - decrements the parameter;
- MINimum - sets the parameter to the smallest possible value;
- MAXimum - sets the parameter to the largest possible value.

1.6.2 Unit Suffixes

Numeric parameters may be followed by an optional suffix:

Suffix	Multiplier
A	1e-18
F	1e-15
P	1e-12
N	1e-9
U	1e-6
M*	1e-3
K	1e3
MA	1e6
G	1e9
T	1e12
PE	1e15
EX	1e18

* - suffix M relates to 1e6 instead of 1e–3 when using MHZ or OHM units.

If the suffix is omitted, default units are used. Various frequency parameters may contain following suffixes:

Suffix	Multiplier
Hz	1e
KHz	1e3
MHz	1e6
GHz	1e9

1.6.3 Boolean Parameters (<boolean>)

Boolean parameters represent a single binary condition that is either true or false. The two-state boolean parameter has four arguments. The following list shows the arguments for the two- state boolean parameter:

- ON or 1 – boolean true;
- OFF or 0 – boolean false.

Boolean parameters are always returned as 1 or 0 by query commands, e.g.:

[SENSe]:AVERage[:STATe] ON|1|OFF|0

will respond with 0 or 1.

1.6.4 Discrete Parameters (<character_data>)

Discrete parameters use mnemonics to represent each valid setting. They have a long and a short form, just like command mnemonics. You can mix upper and lower case letters for discrete parameters:

TRIGger:SOURce {BUS|INTernal|IMMediate|EXTernal}

"BUS", "INTernal", "IMMediate", "EXTernal" are allowed values.

Discrete parameters are always returned in short upper case form.

1.6.5 String Parameters (<string>)

String parameters allow ASCII strings to be sent as parameters. Single or double quotes are used as delimiters, e.g.:

MEMory:ADC:SELect "table_1"

1.7 Queries

All commands, unless otherwise noted, have an additional query form. As defined in IEEE 488.2, a query is a command header with a question mark symbol appended. When a query form of a command is received, the current setting associated with the command is placed in the output buffer. The command and associated query are listed below:

:INP:ATT 20
:INP:ATT?

1.8 Program Message Terminator

LF (0x0A, new line, «\n») symbol (ASCII) is using as program message terminator. «\r\n» (0x0D, 0x0A - new line + carriage return) may be used too, but LF is always returned in query commands.

1.9 The Syntax Conventions

The syntax conventions that are used for all SCPI command keywords and data parameter descriptions in this manual are described below:

- Angle brackets (< >) around a word (or words) indicate they are not to be used literally in the command. They represent the needed item.
- A vertical stroke (|) between keywords or parameters indicates alterative choices. For parameters, the effect of the command varies depending on the choice.
- Square brackets ([ ]) indicate that the enclosed keywords or parameters are optional when composing the command. These implied keywords or parameters will be executed even if they are omitted.
- Braces ({ }) indicate that parameters can optionally be used in the command once, several times, or not at all.

2. IEEE 488 Common Commands

2.1. *CLS

(Write-only) Clears the instrument status byte by emptying the error queue and clearing all event registers. Also cancels any preceding *OPC command or query.

2.2. *IDN?

(Read-only) Returns a string that uniquely identifies the analyzer. The string is of the form *Micran,<model number>,<serial number>,<software version>*.

2.3. *OPC

(Write or Read) Returns an ASCII "+1" when all pending overlapped operations have been completed.

2.4. *RST

(Write-only) Executes a device reset and cancels any pending *OPC command or query, exactly the same as a SYSTem:PRESet.

2.5. *STB?

(Read-only) Reads the value of the instrument status byte. The register is cleared only when the registers feeding it are cleared.

3. ABORt commands

(Write-only) Stops all sweeps - then resume per current trigger settings. This command is the same as INITiate:IMMediate (restart) except if a channel is performing a single sweep, ABORt will stop the sweep, but not initiate another sweep.

4. CALCulate commands

4.1. CALCulate:DATA <char> ,<data>

(Write or Read) Reads Measurement data, read or writes memory data. Data format (text / binary) is defined by FORMat[:DATA]

Parameters

<char> Allowed values:

- **FDATA** - formatted measurement data (query only)
- **FMEM** - formatted measurement data for memory trace import (write only)

<data>

Examples

Read measured data:

```
CALCulate:PARameter:SELect "Trc1"
CALCulate:DATA? FDATA
```

Write ASCII data to a memory trace:

```
CALCulate:PARameter:SELect "Mem1"
CALCulate:DATA FMEM,y1,y2,y3,y4
```

4.2. CALCulate:FORMat <char>

(Write or Read) Sets the display format for the selected trace.

Parameters

<char> Display format.

Allowed values:

- **MW** - mW
- **DBM** - dBm
- **W** - W
- **DBW** - dBW
- **DBV** - dBV
- **DBMV** - dBmV

DBUV - dBuV
- **V** - V
- **MV** - mV
- **UV** - uV
- **NV** - nV
- **DBMHz** - dBm/Hz

4.3. CALCulate:MARKer

Markers operations.

4.3.1. CALCulate:MARKer:AOFF

(Write-only) Reset all markers.

4.4. CALCulate:MARKer<n>

Marker's settings.

Suffix

<n> Marker's number.

4.4.1. CALCulate:MARKer<n>:DISCrete <bool>

(Write or Read) Turns ON / OFF marker's discrete mode.

Parameters

<bool> Discrete mode.

Allowed values:

- **ON**
- **OFF**

4.4.2. CALCulate:MARKer<n>:FUNCtion

Configure marker's function.

4.4.2.1. CALCulate:MARKer<n>:FUNCtion:DOMain

Configures search range of the marker's function.

4.4.2.1.1. CALCulate:MARKer<n>:FUNCtion:DOMain:USER

Configures custom search range of the marker's function.

4.4.2.1.1.1. CALCulate:MARKer<n>:FUNCtion:DOMain:USER[:RANGe] <num>

(Write or Read) Search range selection.

Parameters

<num> Range number.

Allowed values:

- **0** - search entire range of the trace
- **1** - user search range defined by STARt and STOP values

4.4.2.1.1.2. CALCulate:MARKer<n>:FUNCtion:DOMain:USER:STARt <num>

(Write or Read) Start frequency of the search range.

Parameters

<num> Range start.

Supported units: HZ(default), KHZ, MHZ, GHZ, THZ

4.4.2.1.1.3. CALCulate:MARKer<n>:FUNCtion:DOMain:USER:STOP <num>

(Write or Read) Stop frequency of the search range.

Parameters

<num> Range stop.

Supported units: HZ(default), KHZ, MHZ, GHZ, THZ

4.4.2.2. CALCulate:MARKer<n>:FUNCtion[:SELect] <char>

(Write or Read) Select marker's search function.

Parameters

<char> Markers's function.

Allowed values:

- **OFF** - search function inactive
- **MAXimum** - maximum search
- **MINimum** - minimum search
- **TARGet** - search value specified with CALCulate:MARKer:TARGet

4.4.2.3. CALCulate:MARKer<n>:FUNCtion:TRACking <bool>

(Write or Read) Turns ON / OFF tracking. Marker's function is not active if tracking is turned off.

Parameters

<bool> Tracking state.

Allowed values:

- **ON**
- **OFF**

4.4.2.3.1. CALCulate:MARKer<n>:FUNCtion:TRACking:SOURce <Tname>

(Write or Read) Links the marker to a trace to operate with search/tracking functions.

Parameters

<Tname> Trace name.

4.4.3. CALCulate:MARKer<n>[:STATe] <bool>

(Write or Read) Enable / disable marker. It's not possible to retrieve data or process function when marker is disabled.

Parameters

<bool> Marker's state.

Allowed values:

- **ON**

- **OFF**

4.4.4. CALCulate:MARKer<n>:TARGet <num>

(Write or Read) Target value for a search fucntion.

Parameters

<num> Target value to search.

4.4.5. CALCulate:MARKer<n>:TYPE <char>

(Write or Read) Selects marker type - normal / fixed. Fixed marker stores last values of stimulus and traces data.

Parameters

<char> Marker's type.

Allowed values:

- **NORMal**
- **FIXed**

4.4.6. CALCulate:MARKer<n>:X <num>

(Write or Read) Sets the marker's X-axis value (frequency, power, or time).

Parameters

<num> Marker's position.

Supported units: HZ(default), KHZ, MHZ, GHZ, THZ

4.4.7. CALCulate:MARKer<n>:Y? <char>

(Read-only) Query marker's value by the trace name.

Parameters

<char> Trace name.

4.5. CALCulate:PARameter

Lists, creates, selects and deletes traces.

4.5.1. CALCulate:PARameter:CATalog?

(Read-only) Returns the names and measurements of all traces.

Examples

Query result:

```
"Trc1,Power"
```

4.5.2. CALCulate:PARameter[:DEFine] <char> ,<char>

(Write-only) Create data trace.

Parameters

<char> Trace name.

<char> Trace measurement.

Allowed values:

- **POWer**

Examples

Spectrum trace creation:

```
:CALCulate:PARameter:DEFine "Trc1",POWer
```

4.5.3.1. CALCulate:PARameter:DELete:ALL

(Write-only) Deletes all traces.

4.5.3.2. CALCulate:PARameter:DELete[:NAME] <char>

(Write-only) Deletes the specified trace.

Parameters

<char> Trace name.

4.5.4. CALCulate:PARameter:SELect <char>

(Write or Read) Sets the selected trace. Most CALC: commands require that this command be sent before a setting change is made. One trace on each channel can be selected at the same time. To obtain a list of currently named traces, use CALCulate:PARameter:CATalog?.

Parameters

<char> Trace name.

4.6. CALCulate:SMOothing

Trace smoothing function.

4.6.1. CALCulate:SMOothing:APERture <num>

(Write or Read) Sets size of the smoothing window in percents of span.

Parameters

<num> Window size.

Allowed values:

- **MINimum**
- **MAXimum**

Value range: **1** ÷ **30**

Default value: **5**

4.6.2. CALCulate:SMOothing[:STATe] <bool>

(Write or Read) Enable / disable trace smoothing.

Parameters

<bool> Function state.

Allowed values:

- **ON**
- **OFF**

5. DISPlay commands

Configures display setttings.

5.1. DISPlay:CATalog?

(Read-only) Returns the existing window numbers.

5.2. DISPlay:VISible <bool>

(Write or Read) Makes the Graphit application visible or not. In the invisible state, the analyzer cycle time for making measurements, and especially data transfer, can be significantly faster because the display does not process data.

Parameters

<bool> Graphit visibility.

Allowed values:

- **ON**
- **OFF**

5.3. DISPlay:WINDow<wnum>

Controls diagram window.

Suffix

<wnum> The number of a diagram window

5.3.1. DISPlay:WINDow<wnum>:ACTivate

(Write-only) Activates specified window.

5.3.2. DISPlay:WINDow<wnum>:CATalog?

(Read-only) Returns the trace numbers for the specified window.

5.3.3. DISPlay:WINDow<wnum>[:STATe] <bool>

(Write or Read) Write to create or delete a window or read whether a window is present.

Parameters

<bool> The state of a diagram window.

Allowed values:

- **ON**
- **OFF**

5.3.4. DISPlay:WINDow<wnum>:TRACe<tnum>

Trace display parameters.

Suffix

<tnum> Trace number on the specified diagram window.

5.3.4.1. DISPlay:WINDow<wnum>:TRACe<tnum>[:STATe] <bool>

(Write or Read)

Parameters

<bool> Trace visibility

Allowed values:

- **ON**
- **OFF**

5.3.4.2.1.1. DISPlay:WINDow<wnum>:TRACe<tnum>:Y[:SCALe]:AUTO

(Write-only) Performs automatic trace scaling.

5.3.4.2.1.2. DISPlay:WINDow<wnum>:TRACe<tnum>:Y[:SCALe]:PDIVision

(Write or Read) Specifies trace scale division.

Parameters

 Scale division

5.3.4.2.1.3. DISPlay:WINDow<wnum>:TRACe<tnum>:Y[:SCALe]:RLEVel

<ref>

(Write or Read) Specifies trace reference level. Defined by the analyzer's reference level (SENSe:RLEVel) for the spectrum data traces.

Parameters

<ref> Reference level.

5.3.4.2.1.4. DISPlay:WINDow<wnum>:TRACe<tnum>:Y[:SCALe]:RPOSition <pos>

(Write or Read) Specifies trace reference position. Always equals 10 for the spectrum data traces.

Parameters

<pos> Reference position

Allowed values:

- **MINimum**
- **MAXimum**

Value range: **1** ÷ **10**

6. FORMat commands

6.1. FORMat[:DATA] <char> ,<width>

(Write or Read) Sets the data format for data transfers.

Parameters

<char> Data format.

Allowed values:

- **ASCii** - data transfers in ASCII text format
- **REAL** - data transfers in binary 32-bit or 64-bit floating point numbers

<width> Memory size in bits for a floating point number. Must be defined only for REAL format, 32 and 64 values are allowed.

7. INITiate commands

7.1. INITiate:CONTinuous <bool>

(Write or Read) Specifies whether the SK4M trigger source is set to Internal (continuous) or Manual.

Parameters

<bool> Continuous trigger state.

Allowed values:

- **ON**
- **OFF**

7.2. INITiate[:IMMediate]

(Write-only) Stops the current sweeps and immediately sends a trigger.

8. INPut commands

RF input configuration.

8.1. INPut:COUPling <char>

(Write or Read) Specifies alternating current (AC) or direct current (DC) coupling at the analyzer RF input port. Selecting AC coupling switches in a blocking capacitor that blocks any DC voltage present at the analyzer input. This decreases the input frequency range of the analyzer, but prevents damage to the input circuitry of the analyzer if there is a DC voltage present at the RF input.

Parameters

<char> AC/DC coupling.

Allowed values:

- **DC** - direct current
- **AC** - alternating current

9. INSTrument commands

9.1. INSTrument:CATalog?

(Read-only) Returns a string containing a comma separated list of names of all the installed and licensed measurement modes (applications). These names can only be used with the :INSTrument[:SELect] command.

9.2. INSTrument[:SELect] <char>

(Write or Read) Allows you to select the available measurement applications. Command is NOT SUPPORTED, measurement application is automatically determined by the suffix of VISA address.

Parameters

<char> Measurement application.

Allowed values:

- **NFIGURE** - Noise Figure measurement
- **SA** - spectrum analyzer

10. MMEMory commands

The memory commands control saving and loading instrument states and measurement trace data to the hard drive.

10.1. MMEMory:CREate <name> ,<param>

(Write-only) Creates empty memory trace for a following data import (see CALCulate:DATA).

Parameters

<name> Memory trace name.

<param> Measurement name. Must be specified to define supported data formats.

Allowed values:

- **POWer**

Examples

Memory trace *Mem1* for spectrum:

```
MMEMory:CREate "Mem1",POWer
```

10.2. MMEMory:LOAD

Loads user profile or calibration data from file. Data type will be selected by the file extension if type is not specified.

10.2.1. MMEMory:LOAD[:STATe] <filename>

(Write-only) Loads user profile from file.

Parameters

<filename> Filename.

10.3. MMEMory:STORe

Saves user profile or calibration data to file. Data type will be selected by the file extension if type is not specified.

10.3.1. MMEMory:STORe[:STATe] <filename>

(Write-only) Saves user profile to file.

Parameters

<filename> Filename.

11. SENSe commands

11.1. [SENSe]:BANDwidth

Resolution bandwidth parameters.

11.1.1. [SENSe]:BANDwidth[:RESolution] <num>

(Write or Read) Sets the resolution bandwidth (RBW) of the analyzer. Note: Specify 6366 Hz value to select 6.4 kHz bandwidth.

Parameters

<num> Resolution bandwidth.

Supported units: HZ(default), KHZ, MHZ, GHZ, THZ

Allowed values:

- **MINimum**
- **MAXimum**

11.1.1.1. [SENSe]:BANDwidth[:RESolution]:AUTO <bool>

(Write or Read) Automatic RBW selection depending on a frequency span and span to RBW ratio (#SENSe:FREQuency:SPAN:BANDwidth:RESolution:RATio).

Parameters

<bool> Auto RBW.

Allowed values:

- **ON**
- **OFF**

11.1.2. [SENSe]:BANDwidth:VIDeo <num>

(Write or Read) Sets the analyzer post-detection filter (VBW).

Parameters

<num> Video filter bandwidth.

Supported units: HZ(default), KHZ, MHZ, GHZ, THZ

Allowed values:

- **MINimum**
- **MAXimum**

11.1.2.1. [SENSe]:BANDwidth:VIDeo:AUTO <bool>

(Write or Read) Automatic VBW selection depending on a selected RBW and VBW to RBW ratio.

Parameters

<bool> Auto VBW.

Allowed values:

- **ON**
- **OFF**

11.1.2.2. [SENSe]:BANDwidth:VIDeo:RATio <num>

(Write or Read) Selects the ratio between the video bandwidth and the equivalent 3 dB resolution bandwidth to be used for setting VBW when VBW is in Auto.

Parameters

<num> VBW to RBW ratio

Allowed values:

- **MINimum**
- **MAXimum**

Value range: **1e-6** ÷ **100**

Default value: **1**

11.2.1. [SENSe]:DETector[:FUNCtion] <char>

(Write or Read) Selects mode of the display detector.

Parameters

<char> Display detector mode

Allowed values:

- **SAMPle** - sample
- **NORMal** - normal
- **POSitive** - positive peak
- **NEGative** - negative peak
- **AVERage** - RMS average

11.3.1. [SENSe]:FREQuency:CENTer <num>

(Write or Read) Sets the center frequency of the analyzer.

Parameters

<num> Center frequency

Supported units: HZ(default), KHZ, MHZ, GHZ, THZ

Allowed values:

- **MINimum**
- **MAXimum**

11.3.2. [SENSe]:FREQuency[:CW] <num>

(Write or Read) Sets the Continuous Wave (or Fixed) frequency. Must also send SENSe:SWEep:TYPE CW to put the analyzer into CW sweep mode.

Parameters

<num> Fixed frequency

Supported units: HZ(default), KHZ, MHZ, GHZ, THZ

Allowed values:

- **MINimum**
- **MAXimum**

11.3.3. [SENSe]:FREQuency:FIXed <num>

(Write or Read) Sets the Continuous Wave (or Fixed) frequency. Must also send SENSe:SWEep:TYPE CW to put the analyzer into CW sweep mode.

Parameters

<num> Supported units: HZ(default), KHZ, MHZ, GHZ, THZ

Allowed values:

- **MINimum**
- **MAXimum**

11.3.4. [SENSe]:FREQuency:SPAN <num>

(Write or Read) Sets the frequency span of the analyzer.

Parameters

<num> Frequency span

Supported units: HZ(default), KHZ, MHZ, GHZ, THZ

Allowed values:

- **MINimum**
- **MAXimum**

11.3.4.1.1.1. [SENSe]:FREQuency:SPAN:BANDwidth[:RESolution]:RATio <num>

(Write or Read) Selects the ratio between span and resolution bandwidth.

Parameters

<num> Span to RBW ratio.

Allowed values:

- **MINimum**
- **MAXimum**

Value range: **1** ÷ **20000**

Default value: **106**

11.3.5. [SENSe]:FREQuency:STARt <num>

(Write or Read) Sets the start frequency of the analyzer.

Parameters

<num> Start frequency.

Supported units: HZ(default), KHZ, MHZ, GHZ, THZ

Allowed values:

- **MINimum**
- **MAXimum**

11.3.6. [SENSe]:FREQuency:STOP <num>

(Write or Read) Sets the stop frequency of the analyzer.

Parameters

<num> Stop frequency.

Supported units: HZ(default), KHZ, MHZ, GHZ, THZ

Allowed values:

- **MINimum**
- **MAXimum**

11.3.7.1.1. [SENSe]:FREQuency:SYNThesis:AUTO[:STATe] <bool>

(Write or Read) Selects the LO phase noise behavior to optimize dynamic range and speed for various instrument operating conditions.

Parameters

<bool> LO automatic mode.

Allowed values:

- **ON**

- **OFF**

11.3.7.2. [SENSe]:FREQuency:SYNThesis[:STATe] <char>

(Write or Read) Selects the LO (local oscillator) phase noise behavior for various desired operating conditions.

Parameters

<char> Phase noise optimization.

Allowed values:

- **FAST** - fast tuning
- **WIDE** - wide-offset
- **CLOSe** - close-offset

11.4.1. [SENSe]:POWer[:RF]

Controls RF input.

11.4.1.1. [SENSe]:POWer[:RF]:GAIN

RF signal amplification settings.

11.4.1.1.1. [SENSe]:POWer[:RF]:GAIN[:STATe] <bool>

(Write or Read) Controls the internal low-noise amplifier.

Parameters

<bool> LNA state.

Allowed values:

- **ON**
- **OFF**

Default value: **OFF**

11.4.1.2. [SENSe]:POWer[:RF]:PADJust <num>

(Write or Read) Allows you to manually adjust the preselector filter frequency to optimize

its response to the signal of interest.

Parameters

<num> Preselector offset.

Allowed values:

- **MINimum**
- **MAXimum**

Value range: **-32768 ÷ 32767**

Default value: **0**

11.4.2. [SENSe]:POWer:RLEVel <num>

(Write or Read) Speceifies the reference level of the analyzer.

Parameters

<num> Reference level.

Supported units: DBM(default)

Allowed values:

- **MINimum**
- **MAXimum**

11.4.2.1. [SENSe]:POWer:RLEVel:ATTenuation

Controls attenuation parameters.

11.4.2.1.1. [SENSe]:POWer:RLEVel:ATTenuation:LIMit <num>

(Write or Read) Sets minimum RF attenuator value.

Parameters

<num> Minimum RF attenuation.

Supported units: DB(default)

Allowed values:

- **MINimum**
- **MAXimum**

Value range: **0** ÷ **70**

Default value: **0**

11.4.2.1.2. [SENSe]:POWer:RLEVel:ATTenuation:MODE <char>

(Write or Read) Reference level mode.

Parameters

<char> Reference level mode.

Allowed values:

- **NOISe** - measure start
- **OPTimal** - optimal mode
- **FRF** - fixed RF attenuator

11.5. [SENSe]:ROSCillator

Reference oscillator settings.

11.5.1.1. [SENSe]:ROSCillator:EXTernal:FREQuency <num>

(Write or Read) Tells the analyzer the frequency of the external reference.

Parameters

<num> External reference frequency.

Supported units: HZ(default), KHZ, MHZ

Allowed values:

- **MINimum**
- **MAXimum**

11.5.2. [SENSe]:ROSCillator:SOURce <char>

(Write or Read) Defines source of the reference frequency.

Parameters

<char> Reference oscillator's type.

Allowed values:

- **INTernal** - internal
- **EXTernal** - external
- **AUTO** - automatic selection

11.6. [SENSe]:SWEep

Specifies the sweep functions of the analyzer.

11.6.1. [SENSe]:SWEep:POINts <num>

(Write or Read) Sets the number of data points for the measurement.

Parameters

<num> Points number.

Allowed values:

- **MINimum**
- **MAXimum**

Value range: **1** ÷ **10001**

Default value: **501**

11.6.2.1. [SENSe]:SWEep:TRIGger:MODE <char>

(Write or Read) Sets and reads the trigger mode for the specified channel. Specifies "Trigger IN" parameter.

Parameters

<char> Trigger IN mode.

Allowed values:

- **POINt** - measure start
- **SWEep** - sweep start

NPOInt - switch to next point

12. SYSTem commands

Controls and queries settings that affect the SK4M system.

12.1. SYSTem:ERRor?

(Read-only) Returns the next error in the error queue. Each time the analyzer detects an error, it places a message in the error queue.

12.1.1. SYSTem:ERRor:COUNt?

(Read-only) Returns the number of errors in the error queue.

12.2.1. SYSTem:MCLass:CATalog?

(Read-only) Returns list of available optional measurements.

12.3. SYSTem:SET

(Write-only) Sets instrument state from a file, equals to profile's loading MMEMory:LOAD[:STATe].

SCPI Errors Description

Error Code	Error String	Description
(+)0	"No error"	No error
-108	"Parameter not allowed"	More parameters were received than expected for the command. You may have entered an extra parameter, or added a parameter to a command that does not accept a parameter
-109	"Missing parameter"	Fewer parameters were received than expected for the command. You may have omitted one or more parameters that are required for this command
-113	"Undefined header"	A command was received that is not valid for the instrument. You may have misspelled the command, it may not be a valid command, or you may have the wrong interface selected. If you are using the short form of the command, remember that it may contain up to four letters
-114	"Header suffix out of range"	Indicates the value of a header suffix attached to a program mnemonic makes the header invalid.
-121	"Invalid character in number"	Indicates an invalid character for the data type being parsed was encountered. For example, an alpha in a decimal numeric or a "9" in octal data.
-123	"Exponent too large"	Indicates the magnitude of an exponent was greater than 32000, see IEEE 488.2, 7.7.2.4.1.
-128	"Numeric data not allowed"	Indicates that a legal numeric data element was received, but the device does not accept one in this position for the header.
-131	"Invalid suffix"	A suffix was incorrectly specified for a numeric parameter. You may have misspelled the suffix. For example, SENS:FREQ 200KZ
-138	"Suffix not allowed"	A suffix was received following a numeric parameter which does not accept a suffix. For example, INIT:CONT 0Hz
-160	"Block data error"	This error, as well as errors -161 through -169, are generated when parsing a block data element. This particular error message is used if the device cannot detect a more specific error.
-161	"Invalid block data"	Indicates a block data element was expected, but was invalid, see IEEE 488.2, 7.7.6.2. For example, and END message was received before the end length was satisfied.
-168	"Block data not	Indicates a legal block data element was encountered, but not

	allowed"	allowed by the device at this point in parsing.
-211	"Trigger ignored"	Indicates that GET, *TRG or TRIG:IMM was received but was ignored because the device was not in the wait-for-trigger state
-213	"Initiation ignored"	Init ignored.
-221	"Settings conflict"	This message means that command has been received and processed succefully, but will be ignored due to conflicting settings
-222	"Data out of range"	A numeric parameter value is outside the valid range for the command
-224	"Illegal parameter value"	Illegal parameter value. A discrete parameter was received which was not a valid choice for the command. You may have used an invalid parameter choice. For example, TRIG:SOUR EX
-226	"List not same length"	This occurs when frequency and dwell time lists do not correspond in length, for example
-227	"CALC measurement selection set to none"	No trace is selected. Use CALCulate:PARameter:SELect to define selected trace.
-240	"Hardware error"	Indicates that a legal program command or query could not be executed because of a hardware problem in the device.
-300	"Device-specific error"	This event bit (Bit 3) indicates that a device operation did not properly complete due to some condition, such as overrange see IEEE 488.2, 11.5.1.1.6.
-310	"System error"	System error
-320	"Storage fault"	Indicates that the firmware detected a fault when using data storage. This is not an indication of physical damage or failure of any mass storage element.
-350	"Queue overflow"	The error queue is full and another error has occurred which could not be recorded
-360	"Communication error"	This is the generic communication error for devices that cannot detect the more specific errors described for error -361 through -363.
-365	"Time out error"	Operation timeout.